

Publicación práctica para usuarios de

commodore

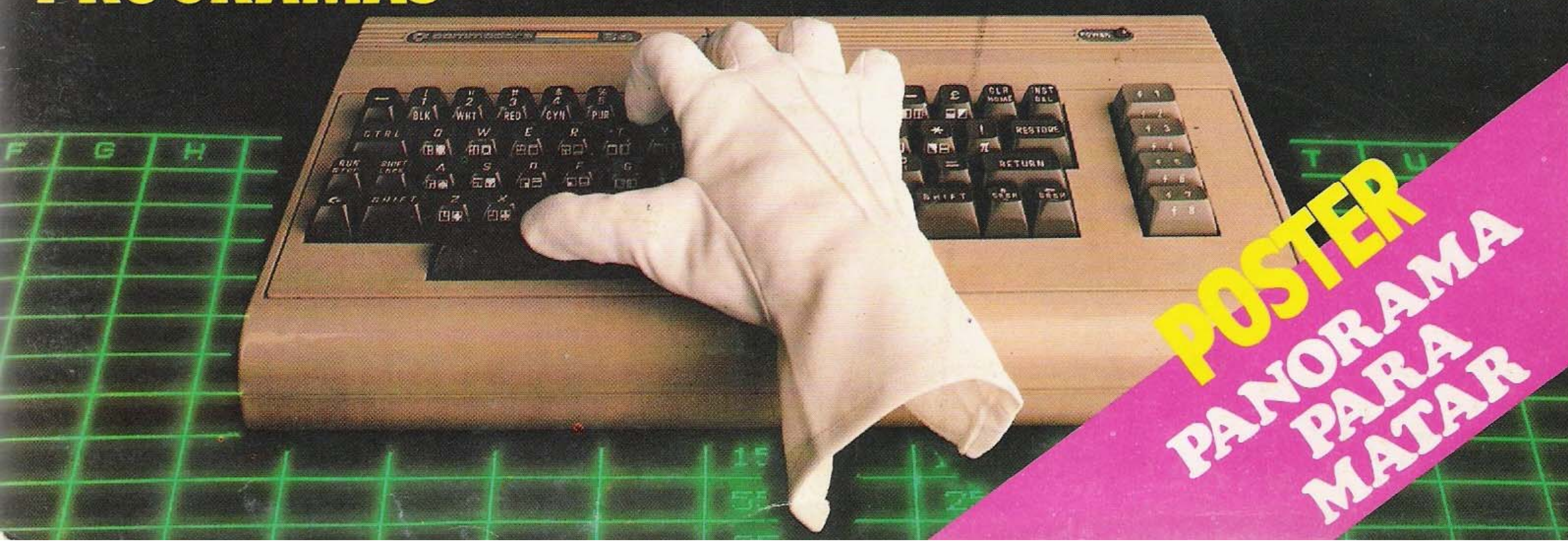
Octubre 1985 Precio 350 Ptas Año 1 Número 2

Contiene un coleccionable

AÑADE COMANDOS GRAFICOS

LA HOJA DE TRABAJO

PROTEGE TUS PROGRAMAS



LOAD'N'RUN

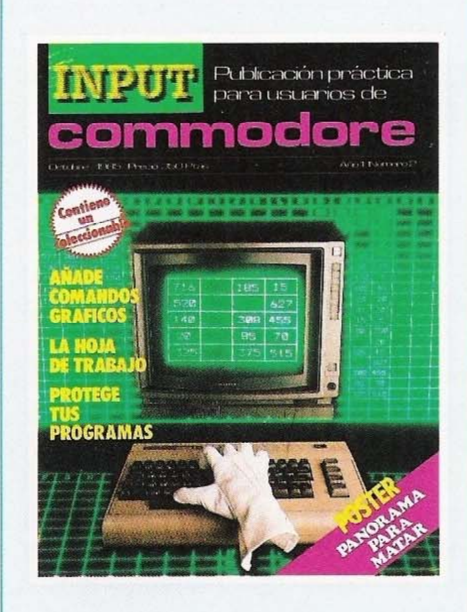
RECOPILACION DE PROGRAMAS EN CASSETTE PARA TU ORDENADOR

LOS MEJORES PROGRAMAS DE EUROPA PARA SPECTRUM Y COMMODORE



Nº1 EN DIFUSION EN CALIDAD DE CONTENIDO EN PRESENTACION EN PRECIO

Mora



DIRECTOR:
Alejandro Diges
COORDINADOR EDITORIAL:
Francisco de Molina
DISEÑO GRAFICO:
Tomás López
COLABORADORES:
Antonio Taratiel, Luis R. Palencia, Christophe Pais, Francisco Tórtola, José Pérez Montero, Benito Román
INPUT Commodore es una publicación juvenil de EDICIONES FORUM
GERENTE DIVISION DE REVISTAS:
Angel Sabat

PUBLICIDAD: Grupo Jota
Madrid: c/ Cuenca, 1, 1.º
Tel. 253 45 01/02
Barcelona: Avda. de Sarriá, 11-13, 1.º
Tel. 250 23 99
FOTOMECANICA: Ochoa, S.A.
COMPOSICION: Fernández Ciudad, S.L.
IMPRESION: Edime, S.A.
Depósito legal: M. 27.884-1985
Suscripciones: EDISA,
López de Hoyos, 141. 28002 Madrid
Tel. (91) 415 97 12
Redacción:
Alberto Alcocer, 46, 4.º
28016 Madrid. Tel. 250 10 00
DISTRIBUIDORA:
R.B.A. PROMOTORA DE EDICIONES, S. A.
Travesera de Gracia, 56. Edificio Odiseus.
08006 Barcelona
Se solicitará el control OJD

INPUT commodore

Sumario

AÑO 1 NUMERO 2

VUESTRAS CARTAS

Parece que hemos entrado con buen pie. Al menos así lo indica el gran volumen de cartas recibidas. Ocupa el primer lugar, por orden de importancia, los envíos a **Los mejores de INPUT.** Tal es así que aumentaremos la cantidad de cintas a sortear. En el próximo número publicamos la lista de premiados, aunque las cintas ya están en camino.

Vuestros artículos y programas, para la sección **Participa,** comenzaron a llegar tímidamente los primeros días, pero acaba de producirse una verdadera avalancha.

Enviáis cartas de contenido realmente halagador en un noventa y nueve por ciento, pero hay un uno por ciento que incorporan una actitud de «¿Qué quieren vendernos estos señores?», vamos a ver en qué les cazamos». Pues bien, es lógico que haya pequeños fallos, achacables a la puesta en marcha de la revista, pero si se miran con lupa se ven más grandes. No olvidéis que escribimos para y no contra vosotros. En realidad, el equipo de redacción está formado por gente tan entusiasta de los micros como podáis serlo cualquiera.

El precio ha sido el tema de alguna carta. En realidad no está tan desfasado con respecto a otras revistas nacionales de ésta y otras especialidades. Por otro lado, las revistas extranjeras, que abundan en los kioscos, no son precisamente baratas. La calidad de impresión tiene sus costos y hemos ajustado los márgenes al máximo. Siempre ponemos como ejemplo a las publicaciones extranjeras y nuestra primera intención ha sido proporcionaros (a nosotros también) una revista de nivel comparable con ellas. Queremos seguir en ese camino. Vosotros, obligándonos a imprimir una Segunda Edición, parece que estáis igualmente de acuerdo.

BALONCESTO POR ORDENADOR

Si crees que puedes ser el campeón baloncestista informático del año puedes inscribirte para competir.

BASES:

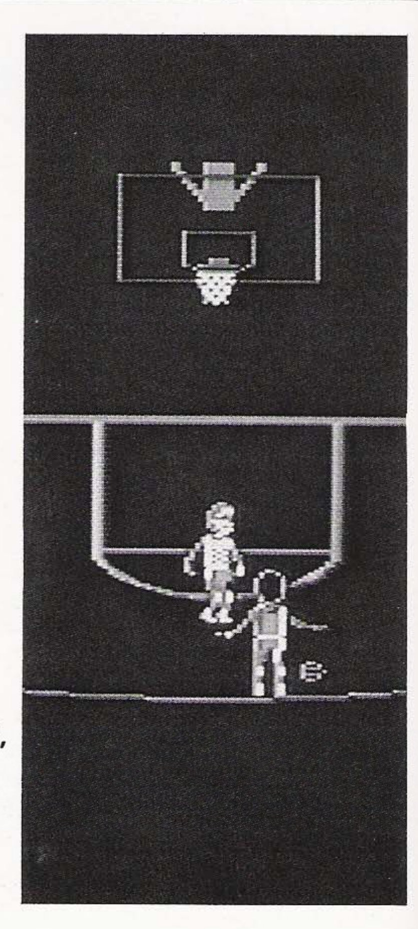
- EL JUEGO UTILIZADO ES EL ONE-ON-ONE, PARA EL COMMODORE 64.
- TENDRA CARACTER ELIMINATORIO.
- LAS PARTIDAS SE CELEBRARAN ENTRE DOS JUGADORES.
- CADA JUGADOR PODRA LLEVAR SU PROPIO JOYSTICK.
- EL LUGAR DE LA COMPETICION ES EL PUB REBOTE (Campoamor, 13. MADRID)

PREMIOS:

- UN ORDENADOR COMMODORE 64
- UN LECTOR DE CASSETTES PARA EL 64
- DIEZ SUSCRIPCIONES A INPUT COMMODORE
- PROGRAMAS DE ONE-ON-ONE
- CAMISETAS
- TROFEOS Y ALGUNAS SORPRESAS MAS

CONVOCAN: INPUT COMMODORE, SERMA Y WINKEL

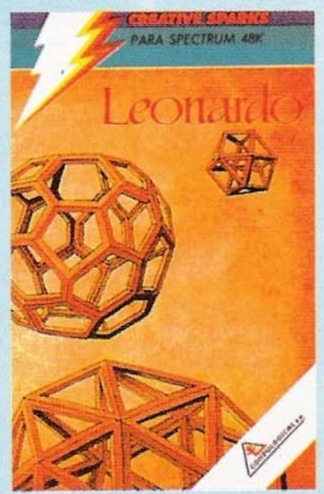
INSCRIBETE LLAMANDO
A INPUT (91) 250 10 00
o SERMA (91) 431 38 11/79





Santa Cruz de Marcenado, 31 - 28015 Madrid - Teléf. 241 10 63

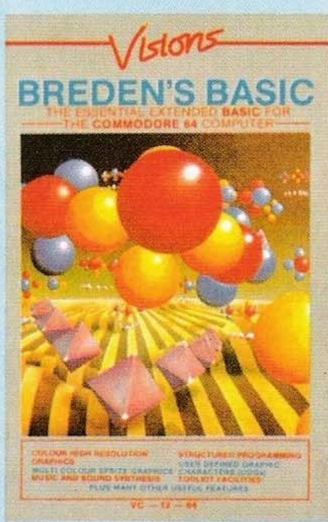
Solicite nuestro amplio catálogo



SPECTRUM

Un paquete gráfico de fácil manejo que puede ser utilizado por artistas amateurs, delineantes o programadores que deseen realizar sus programas con estupendos gráficos y sofisticadas técnicas gráficas.

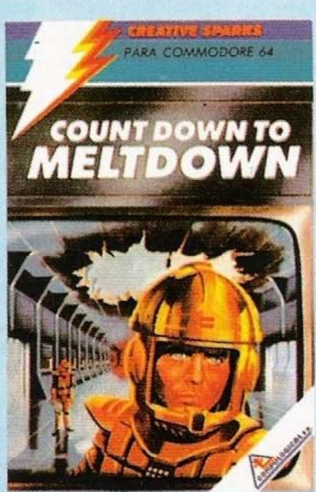
Leonardo contiene más comandos que cualquiera de sus rivales. El programa incluye una completa colección de ejemplos de cómo obtener los mejores resultados de más de 40 comandos.



C-64

La extensión esencial del BASIC para su COMMODORE-64. BREDEN'S BASIC, programa escrito en código máquina aporta 135 nuevos comandos y permite obtener los mejores efectos en sonido y gráficos.
Estos comandos abarcan los siguientes conceptos:

- Manejo de sprites.
- Control de gráficos y pantalla.
- Música y sonido.
- I/O (entrada y salida).
- Disco.
- Manipulación numérica.
- Detección de errores.
- Ayuda al programador.
- Programación estructurada.



C-64 AMSTRAD SPECTRUM

Experimente la trepidante acción del FUTBOL AMERICANO.

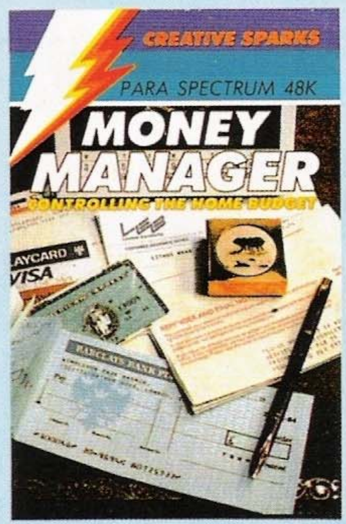
- Para 1 ó 2 jugadores.
- Rápidos gráficos animados.

C-64

Un divertido juego de aventuras y acción que le pone al mando de un equipo de androides intentando alcanzar el centro de un reactor nuclear antes de que explote. Este juego tiene cerca de 2.000 habitaciones para explorar, y cada una de ellas es de 3 dimensiones.

Probablemente el juego más excitante hasta ahora desarrollado para el C-64.

- Rápida carga en cassette.

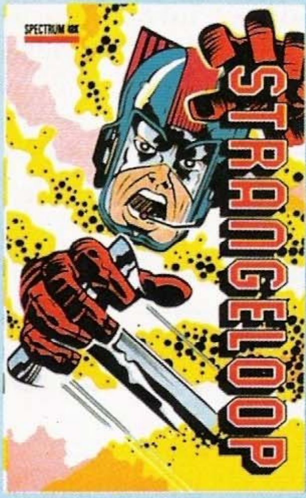


SPECTRUM

Todo el mundo de las finanzas a su alcance por el precio de un juego.

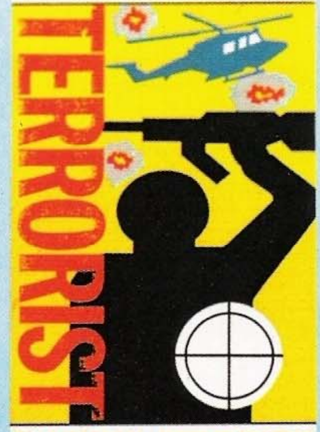
Se puede guardar toda la información en una cinta cassette.

Cualquier información de pantalla se puede sacar impresa con una impresora ZX.



SPECTRUM

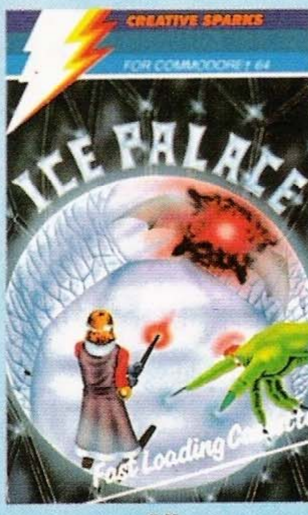
- 240 lugares especialmente detallados, muchos de los cuales contienen sorpresas ocultas y los ingredientes necesarios para completar el juego.
- Robots interactivos.
- Una tremenda aventura de arcada con abundancia de increíbles gráficos.



C-64

Defienda su ciudad contra los terroristas!!

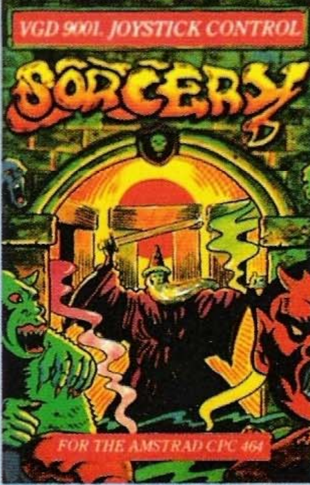
- Divertidas secuencias de batallas en 3D.
- Elección de estrategias multiples.
- Secuencias de ataque constantemente cambiantes.



C-64

Salve a la Tierra del Terrible Hechizo del Hielo!

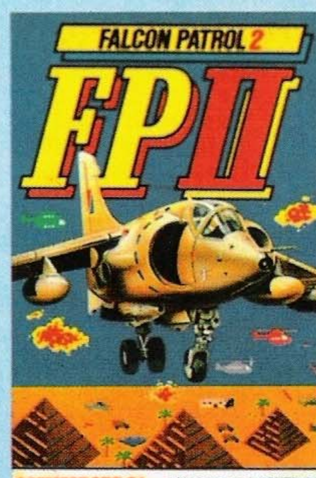
- Acción en tiempo real.
- 1.200 lugares repartidos en 7 niveles de juego.
- Control joystick.



C-64 AMSTRAD SPECTRUM

Atrévase a penetrar en el mundo de las tinieblas!

- Gráficos en 3D magnificamente detallados.
- 17 pantallas para realizar.
- Emocionante test de su coordinación y estrategia en este juego que provoca adicción.



C-64

- Increíbles gráficos en 3 D.
- 16 niveles de juego con dificultad en aumento.
- Interferencia de radar.
- Helicópteros ametralladores.
- Misiles tierra-aire y aire-aire.



AHORA LOS GOONIES

La película de Steven Spielberg, Los Goonies, dará bytes a un nuevo programa que está desarrollando la casa americana Datasoft, para el 64.
La trama del juego consiste en buscar el tesoro del pirata, siendo continuamente incordiados por Mamma Fratelli.
Se espera que esté listo en los próximos días.



LLEGO EPYX

Compulogical, firma importadora y distribuidora de *software*- principalmente para Spectrum y Commodore- acaba de llegar a un acuerdo con uno de los más prestigiosos fabricantes americanos de juegos para ordenador: Epyx.
El resultado más inmediato se traduce en la disponibilidad de juegos tales como Mission Imposible, Winter y Summer Games II, Breakdance, etc., caracterizados por la alta calidad de sus gráficos.



RETRASO DEL C-128

Anunciabamos en esta misma sección, **el mes pasado**, la posibilidad de ver en las tiendas el nuevo Commodore 128 a lo largo septiembre. Sin embargo, el acontecimiento necesitará más tiempo para materializarse.
El principal culpable del retraso parece ser el nuevo arancel, que ha tirado por tierra la estrategia de *marketing* del producto. Ahora está por definir el nuevo precio.



SISTEMA OPERATIVO PARA EL AMIGA

El sistema operativo (conjunto de rutinas escritas en código máquina) da el toque de personalidad a cualquier ordenador. De el depende la eficiencia de la autogestión de los recursos incorporados en su diseño.
El esperado ordenador Amiga responde a un diseño electrónico sofisticado y, por ello, es obligado dotarle de un sistema operativo acorde, para maximizar sus prestaciones.
La compañia que fue contratada originalmente por Commodore para esta misión ha tenido problemas con los tiempos de entrega y es por ello que han decidido confiarsela a la firma británica Metacomco, que está próxima a presentar el TRIPOS/Amiga DOS, que así se llamará el nuevo S.O.
Recordemos que el Amiga incorpora el microprocesador de 16/32 bits 68000 de la firma Motorola.



SISTEMA EXPERTO

Una firma americana, llamada Abacus Software ha lanzado recientemente un sistema experto que utiliza criterios complejos en la búsqueda y gestión de la información, introducida previamente en su base de datos.
El sistema XPER ha sido desarrollado para los ordenadores Commodore 64 y 128.
No obstante, introducir la información es extremadamente sencillo, pudiendo actualizarse siempre que sea necesario.
Igualmente, la generación de informes elaborados tampoco plantea muchas dificultades
Aunque por el momento no se distribuye aquí, esta es la dirección de la firma:
Abacus Software
2201 Kalamazoo S.E.
P.O. Box 7211.
Grand Rapids.
MI 49510.
EE.UU.



COMIENZA LA GUERRA

A mediados de julio, la lucha contra la copia ilegal de *software* recibió un espaldarazo definitivo en Gran Bretaña.
En efecto, una antigua ley de 1956 sobre el *copyright* ha sido rectificada, observandose en ella los derechos sobre el *software* para ordenadores.
En adelante serán severamente castigados quienes sean hallados culpables de duplicar, distribuir o impotar *soft* sin los debidos permisos.
La multa será determinada por el tribunal, sin limitaciones y el culpable podría ir a la carcel hasta dos años.
Los que tengan en su poder o vendan programas "piratas" podrán verse multados con hasta dos mil libras esterlinas por cada delito.



EL MERCENARIO

A decir de quienes han visto los *prereleases*, El Mercenario es un juego que dará mucho que hablar. Es una mezcla de marcianitos, y los conocidos simuladores de vuelo interplanetario y gráficos tridimensionales vectoriales.
Buscas en el cielo y las profundidades del planeta Targ los suministros y cristales energéticos que precisas para ser un eficaz mercenario del siglo XXI.

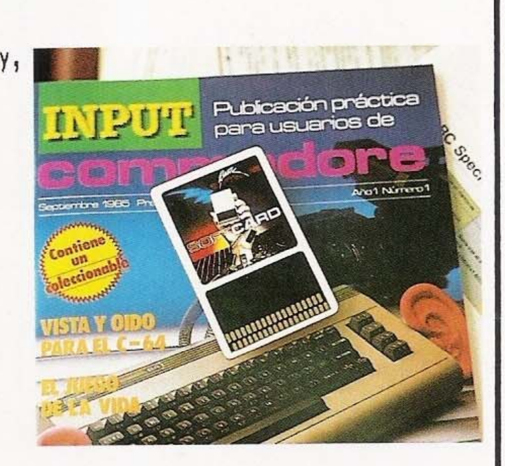


NUEVOS ARANCELES

Un caluroso día de julio algo sobresalto a los importadores de microordenadores que abrieron el Boletín Oficial del Estado.
Una indiscriminada subida arancelaria lineal, de quince mil pesetas, afectaba a todos los sistemas cuyo precio estuviera por debajo de las trescientas mil.
La cosa creaba dudas, hasta el punto de que cualquier periférico o accesorio pudiera sufrir igual trato.
Esta situación se corrige a principios de septiembre, logicamente a través del B.O.E.
Indignación e impotencia es el sentimiento que inunda a una mayoría de los atónitos distribuidores.
Los intentos para hallar justificación a ésta medida apuntan a la clásica protección de una industria nacional. Es escualida, por no decir nula, argumentan algunos y la especulación hace pensar en la llegada de fabricantes extranjeros, entre los que se barajan Fujitsu, de la mano de Secoinsa, y su modelo F-7. Tal vez sea Sinclair o Atari, dicen otros. Incluso hay quienes apuntan directamente a Dragon y su nuevo proyecto en Galicia como detonante.



LA TARJETA CREDISOFT



Parece una tarjeta de crédito y, sin embargo, es portadora de sueños. En su interior alberga un diminuto *chip* de ROM (memoria que solo puede leerse) en el que se ha dispuesto información, que muy bien puede ser el código máquina de un juego.
Al igual que ocurre con los *chips* de ROM clásicos, existen alternativas para las nuevas tarjetas. Una variante es la PROM, que es una memoria en blanco que puede ser grabada por el usuario, gracias a un dispositivo electrónico especial. Otra es la EPROM, en la que pueden ser grabados los datos y posteriormente borrados para almacenar nueva información.
El desarrollo es fruto de la colaboración de dos empresas: Mitsubishi y Hudson Soft.
Las capacidades de la versión grabada por el fabricante cubren un rango situado entre los 64 Kits y 1 Mbit, mientras que las otras dos tienen, por ahora, como límite máximo los 256 Kbits.
Los ordenadores MSX ya disponen de un adaptador, con forma de cartucho, que permite la conexión directa
al bus del sistema. En breve estarán disponibles las versiones para Spectrum y los Commodore.
Las treinta y dos diminutas pistas eléctricas (recubiertas de oro) se conectan a los respectivos *ports* del usuario en el ordenador, por medio del aludido adaptador.
Esta nueva modalidad de presentación del *software* pone mas alta la barrera a los "piratas", que deberán ser mas sofisticados.
El formato de la tarjeta es de ochenta y seis por cincuenta y cuatro milimetros, siendo uno con ocho su espesor.
Además, se puede imprimir un diseño gráfico o texto por ambas caras.
La firma distribuidora Serma ha firmado con Hudson para la comercialización de la Beecard en nuestro pais.



LA MEMORIA DEL 128

Esto es como el cuento de "La lechera", no hace más que aparecer el C-128 en algunos paises y ya se sueña con llevarlo a mas.
Es la propia Commodore quien anuncia cartuchos de expansión de la memoria RAM y que conectados a través del *port* del usuario, harán que el 128 pueda tener hasta 512 Kbytes.
Resulta interesante saber que cuando el ordenador trabaja en modo CP/M-80 la RAM que pasa de 64 K actua como "pseudodisco" de estados sólido; es decir, el microprocesador acude al resto de la memoria tratandola como si fuera una unidad de *diskettes* ultrarápida, con una velocidad de transferencia de datos de aproximadamente un megabyte por segundo.

...7 Personajes, 10 acordes perdidos, 15 horas, 48 guardias de tráfico, 95 estaciones de Metro, 45.000 metros cuadrados de Londres, 7 millones de londinenses,... 943 pantallas con trepidantes acciones.



SOFTW

NAPOLES, 98, 1.º 3

08013 BAR

PAUL McCARTNEY'S *Give my regards to*

BROAD STREET

Cuando la musica acabe, el misterio comenzara...





SPECTRUM 48K Y COMMODORE 64





© 1985



WER

ARE, S.A.

Tels. 232 24 61 - 232 25 52
LONA (SPAIN)



SOFTWARE

CENTURY SOFTWARE

¡¡¡ATENCION!!!
POWER, SOFTWARE, S.A.
Perseguirá por todos los medios legales a su alcance, cualquier tipo de piratería, comercialización, importaciones o duplicados ilegales que sobre sus programas se practique.



FRENZY SPECTRUM GAMES FROM QUICKSILVA

RUNS ON THE 16K/4





BUG-BY
FRI
48K Spectrum
FRENZY

FRENZY

Gráficos en alta resolución a todo color, sonido, tabla de puntuaciones, pausa, cinco niveles de juego, robots, disparos Laser, monstruos asesinos, minas seguidoras explosivas, dificultad progresiva.

QUINTIC WARRIOR

Parte hombre y parte superman, el guerrero Quintic se opone al siniestro hombre cangrejo y a las máquinas mutantes. ¿Estás dispuesto para el combate?

FRIDGE

Quienesquiera que hayan cargado la nave espacial de Freddie han provocado una gran confusión. Ahora Freddie está intentando volver a casa, pero el combustible radiactivo para el regreso se halla dispersado en su frigorífico.

STRONTIUM DOG

Johny Alpha es un Agente Investigador Destructor, un cazador de recompensas del futuro. Johny lleva consigo Electro bengalas, que deslumbrarán a sus enemigos, obligándoles a dejar de disparar, y Bombas de Tiempo. Buena suerte en su Misión.

Deseo recibir los juegos que a continuación especifico, comprometiéndome al pago del importe de los mismos.

Nombre ______

Dirección ______

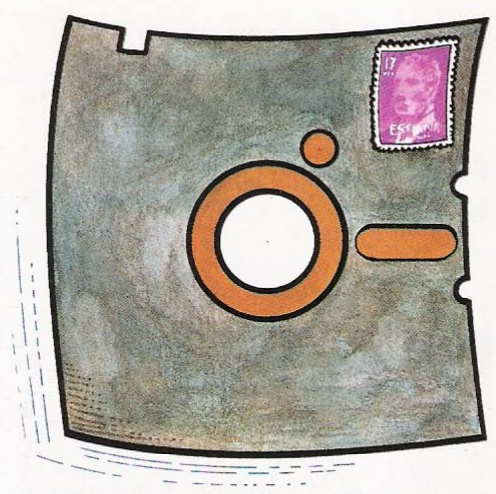
Teléfono ______

Firma:

TITULO	SISTEMA	PRECIO UNIT.	CANTIDAD
LASER ZONE	Spectrum	975 Pts.	
GRIDRUNNER	Spectrum	975 Pts.	
FRENZY	Spectrum	1.275 Pts.	
ASTRO BLASTER	Spectrum	1.275 Pts.	
QUINTIC WARRIOR	Commodore	1.275 Pts.	
PURPLE TURTLES	Commodore	1.275 Pts.	
STRONTIUM DOG	Spectrum	1.800 Pts.	
STRONTIUM DOG	Commodore	2.100 Pts.	
FRIDGE FRENZY	Spectrum	1.800 Pts.	
BROAD STREET	Spectrum	2.400 Pts.	
BROAD STREET	Commodore	2.600 Pts.	

☐ Contrarreembolso. ☐ Adjunto Talón. ☐ Giro Postal.

Deseo recibir información de sus programas en: MSX ☐ AMSTRAD ☐



EL BUZON DE INPUT

Me gustaría que publicáseis el precio aproximado y dónde puedo conseguir (vivo en Valencia) el VIDEO DIGITALIZADOR y el VOICE MASTER.

Una sugerencia: Sería conveniente que en el apartado de código máquina, además de dar programas hechos, diérais un pequeño curso de código orientado hacia los que $C000 nos suena a frigorífico.

Luis. Valencia

Si miras más detenidamente las páginas de publicidad de la revista encontrarás la información que deseas. Una vez localizado el proveedor, contacta con él, es muy probable que disponga de una sección de envíos por correo.

En cuanto a la publicación del precio sucede que en el momento en que dispusimos de los equipos para testearlos, el precio estaba aún por definir.

El código máquina es una de las obsesiones de cuantos desean profundizar cada vez más en la microinformática. Una vez adquiridos los conocimientos básicos resulta casi tan sencillo como programar en BASIC. Es por ello que estamos elaborando una serie de artículos, pero cuidando su nivel didáctico al máximo. ¿Verdad que resulta descorazonador ver que te hablan de los direccionamientos, acumuladores, registros, contador de programa, etc., sin definir exactamente su utilización práctica? Pues nosotros te contaremos todo eso, pero dándole su justo valor en cada momento. De verdad que os gustará y además no va a tardar mucho en asomarse a las páginas de ***INPUT****.*

Soy radioaficionado, y el ordenador lo utilizo muchísimo de cara a la transmisión y recepción RTTY y C. W. Lo tengo conectado a un equipo de 2 m., Azden PCS.2000. El interface, que es lo más necesario, es casero, pero funciona bien.

Desearía listado o grabación en cinta, de otro programa ETTY y C. W., ya que el mío es muy sencillo y no tiene memorias para los mensajes.

EB3-P. P. O. Box de Esplugas de Llobregat.

(Por favor, publicarla, necesito el programa... PSE KKKKKKKK)

Ya lo sabéis, nuestro amigo Xavier quiere invadir el éter con sus 144 Megahertzios, y de paso..., pensar que podríamos crear una sección para vosotros. Escribirnos.

¿Cómo poder ver un programa entero cuando sólo se ve SYS y un número detrás?

José Luis Fernández. Madrid

¿Tú también estás tratando de «abrir» un programa comercial? SYS es un comando que, desde el BASIC, hace que se ejecute un programa desarrollado en código máquina, que comienza en la dirección de memoria indicada por el número al que aludes. Para ver lo que se esconde detrás, puedes coger un cartucho, tal como el Hesmon, y utilizar el comando D para desensamblar a partir de esa dirección. Como curiosidad está bien, pero es tremendamente laborioso.

Quisiera saber si su concurso, que se celebra cada mes, y consta de tres premios en material informático, se realizará mediante un JURADO o por SORTEO; ya que al leer las bases me he encontrado con que hablaban de JURADO y después de SORTEAR premios.

Norbert Rovira. Barcelona

Efectivamente, has leído bien. Hay jurado y sorteo simultáneamente. El jurado decide qué trabajos serán publicados, y la recompensa que recibirá su autor. El sorteo se celebra tomando a todas y cada una de las colaboraciones enviadas durante el mes. Se pretende así que opten a un premio quienes tengan verdadero interés por participar en INPUT*. El esfuerzo merece ser reconocido, independientemente de que se publique o no un trabajo.*

Tengo un C-64 y compré un juego (Buscando mis huesos: Where's my bones?), y no se me cargó, lo llevé a la tienda y, para sorpresa mía, allí se cargó.

Se dice que va a implantarse el sistema MSX y no va a haber otro. ¿Qué pasará con los Sinclair y Commodores? ¿Se quedarán antiguos y suprimidos por los MSX?

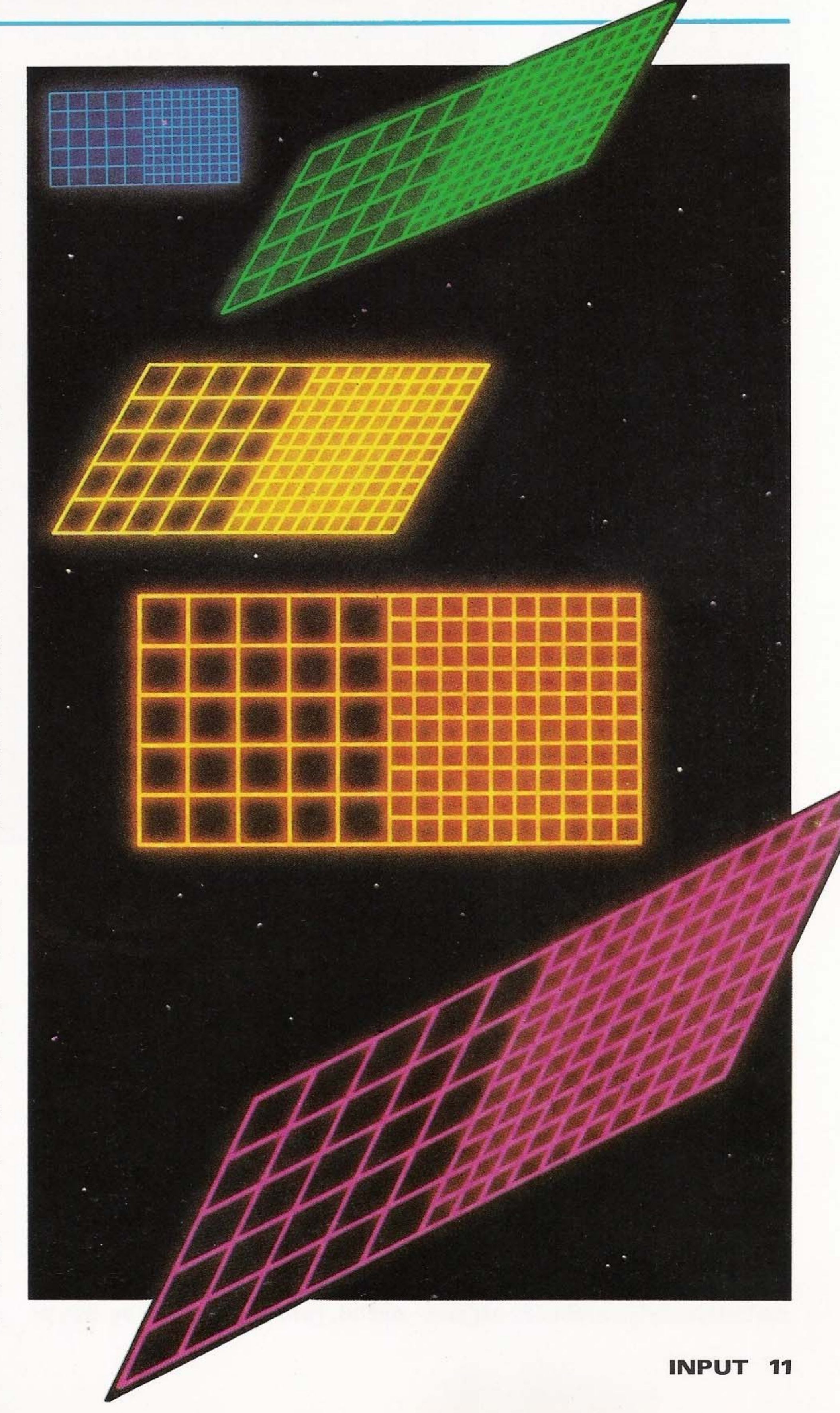
Bernardo Sánchez

El problema no es con tu ordenador, sino con la unidad de cassette, casi con toda seguridad. En la parte superior de la unidad ***C2N*** *verás un pequeño agujerito, por el que puedes introducir la punta de un destornillador cuando el aparato está con la tecla PLAY pulsada. Girando media vuelta a la izquierda o la derecha, es muy probable que ajustes el «azimut», hasta conseguir una buena carga. Si sigues teniendo el mismo problema, acude con esta unidad al servicio técnico.*

El sistema ***MSX*** *fue desarrollado por la casa americana* ***Microsoft****, que posteriormente vendió los derechos a varios fabricantes japoneses interesados en fabricar ordenadores. El fenómeno* ***MSX*** *puede compararse, con grandes diferencias, a lo que pueda significar el sistema* ***VHS*** *en vídeos. Eso no quiere decir que desaparezcan otros, sino que compartirán el mercado. De cualquier forma, el fabricante que tenga gran número de unidades vendidas de sus modelos tiene asegurada su permanencia. La fidelidad del usuario a su marca es un factor clave que los fabricantes saben estimar en lo que vale.*

GRAFICOS EN ALTA RESOLUCION PARA EL COMMODORE 64

Presentamos un programa en código máquina cuya utilidad está fuera de toda duda. Tiene por cometido ampliar el BASIC estándar del 64, proporcionándole comandos directos para gráficos en alta resolución.



En muchos de los programas que aparecerán en INPUT hacemos uso de comandos que sirven para dibujar y pintar en la pantalla, tales como HIRES o COLOUR. Estos son aportados por un cartucho especial, el **Simon's BASIC,** y no están disponibles en el **64** básico, salvo que teclees el programa que presentamos a continuación. Con él obtendrás una equivalencia del cartucho y el pleno aprovechamiento de la revista.

A lo largo de varios artículos iremos enriqueciendo el surtido de comandos y simultáneamente explicaremos cómo funcionan. Así, artículos tales como el publicado en nuestro número anterior, titulado *Así lo oye tu ordenador,* se podrán utilizar fácilmente con tomar la precaución básica de añadir el carácter @ delante del comando gráfico. Es decir, donde aparezca, por ejemplo, HIRES, escribiremos @HIRES.

El programa se irá depositando en las direcciones de memoria comprendidas entre las $C000 y $CFFF (hexadecimales) o 49152 a 53247 (equivalente en decimal).

El programa completo es relativamente largo, por lo que ha sido dividido en módulos funcionales que hacen menos ardua su copia y más detallada la explicación de las líneas de programa.

Originalmente, el programa fue desarrollado con la ayuda de un Ensamblador y después cargado en la memoria del ordenador, pero suponemos que no todos disponéis de uno y los conocimientos para usarlo. Es por ello que presentamos cada rutina

(módulo) en tres versiones equivalentes, a elegir por vosotros mismos cuál utilizar.

La primera es para quienes uteliçéis un ensamblador, tal como el **Macro** o la combinación **HESEDIT, HESBAL,** por poner un ejemplo.

La segunda se presenta en forma de bytes escritos en hexadecimal y agrupados en líneas de a ocho, para teclearlos utilizando algún cartucho del tipo **HESMON;** aunque con este modelo también puedes ensamblar de una forma más primitiva, pero igualmente válida, con ayuda del comando A.

La tercera es la más simple. Son cortos programas en BASIC, que cargan los bytes incluidos en las sentencias DATA en las direcciones de memoria definidas por un bucle FOR-...NEXT. POKE deposita directamente byte (número recogido de las DATA) en la dirección señalada. Estos cortos programitas se copian y ejecutan como si de cualquier otro programa en BASIC se tratara.

Un punto común a cualquiera de los tres métodos que elijáis: No olvidar que debemos cargar en cinta o *diskette* lo que tecleamos. De otra forma se pierde al desconectar el ordenador y habrá que repetir el trabajo de nuevo.

AJUSTES PREVIOS

Supongamos que hemos escrito un programa BASIC cualquiera en el ordenador y que deseamos verlo funcionar. El siguiente paso es simple, teclear RUN. Es en este momento cuando entra en acción una corta rutina, incluida en el ordenador por el fabricante, que lleva por nombre CHRGET *(gecharacter* = consigue el siguiente carácter). Todos sabemos que un programa en BASIC se escribe en forma de una sucesión de líneas semejantes a un texto y que la tecla RETURN hace que cada nueva línea pase a ocupar un espacio en la memoria. Pues bien, CHRGET desmenuza ese texto, leyendo carácter por carácter, para que el programa se ejecute de acuerdo a las órdenes que el mismo incluye. Al principio de cada nueva línea espera encontrar una **palabra clave** del BASIC, tal como PRINT, REM o LET (aunque esta última pueda ser normalmente omitida). Si encuentra algo raro, como @HIRES, escribirá un mensaje de «sintax error», que denuncia que algo no está correctamente escrito.

El primer módulo que presentamos es una corta rutina que cortocircuita parcialmente el funcionamiento de CHRGET y deriva el control del ordenador a nuestro programa. Este verificará si aparece por algún lado el símbolo @ y, si es así, examina cuál es el comando que le sigue. De otro modo, el control es devuelto a las rutinas originales del **64.**

* = $C000 es equivalente a ORG 49152 en otros ensambladores. Si empleáis el **HESMON** comenzaréis escribiendo A$C000 LDX # S02 (RETURN), etc., o bien: $C000 A2 02 BD...

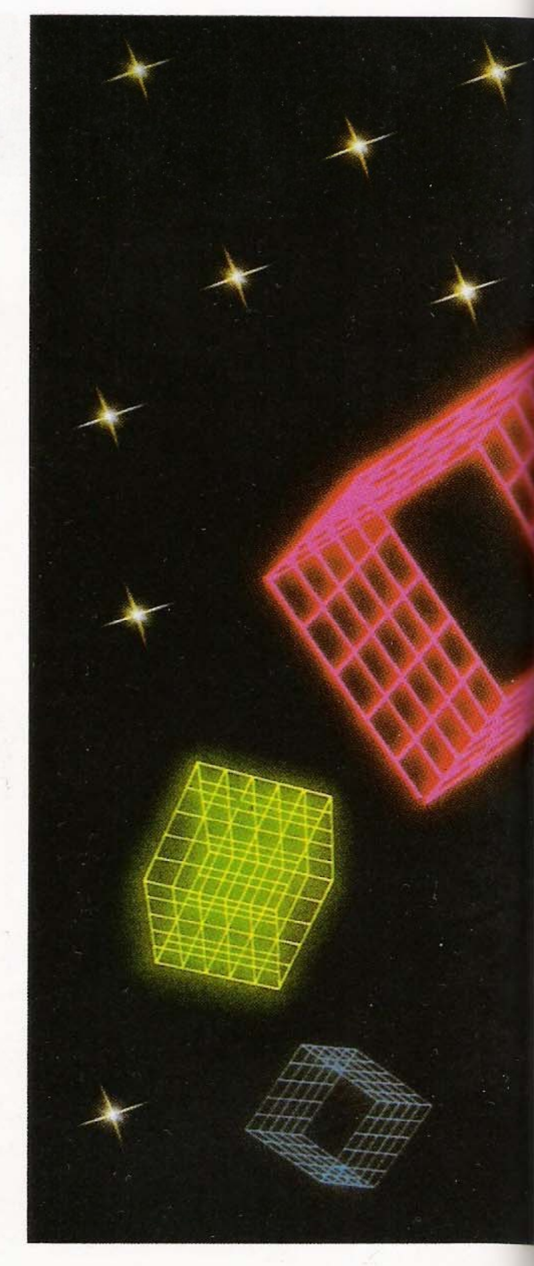
La versión en BASIC no precisa más indicaciones que la necesidad de teclear SYS 49152 para que se ejecute la rutina.

```
*=$C000;ORG49152
       LDX #$02
VUELVE LDA $C00B,X
       STA $73,X
       DEX
       BPL VUELVE
       RTS
       JMP $C00F
       .BYT $00
```

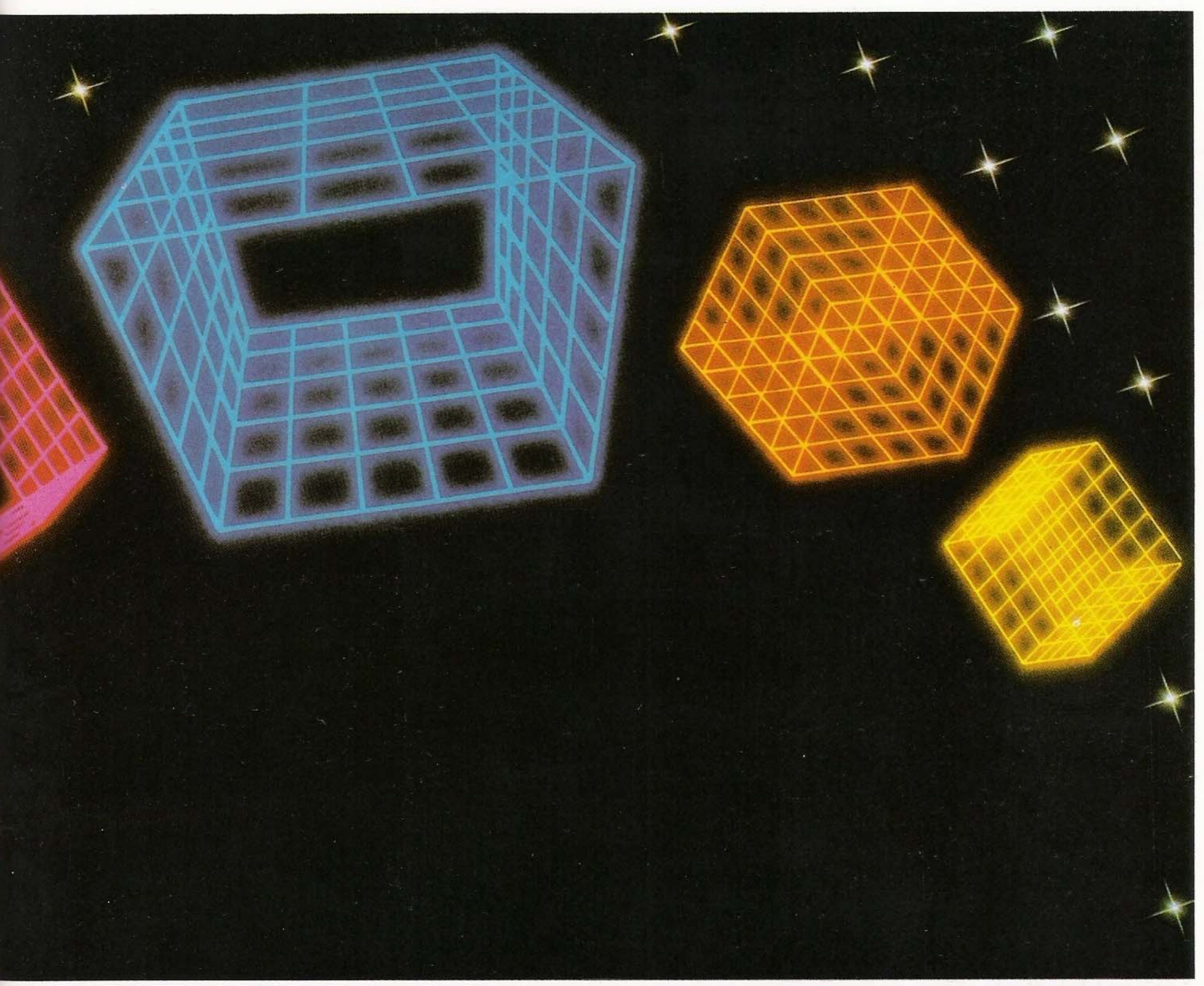
```
:C000 A2 02 BD 0B C0 95 73 CA
:C008 10 F8 60 4C 0F C0 00
```

```
 10 T=0
 20 FOR I=49152 TO 49166
 30 READ A
 40 POKE I,A
 50 T=T+A
 60 NEXT I
 70 IF T<>1565 THEN PRINT
    "ERROR EN LOS DATA"
100 DATA 162,2,189,11,192,
    149,115,202,16,248
110 DATA 96,76,15,92,0
```

CHRGET está incluida en la ROM del **64,** pero es copiada en un área de la RAM, en la página cero, durante los segundos que siguen a la conexión inicial del ordenador. Por estar en la RAM es alterable. Comienza en la dirección $0073 de la memoria. Nuestra rutina se encarga de depositar en las direcciones $73, $74 y $75 los tres bytes correspondientes a la instrucción JMP $C00F, que hará que el control pase a la dirección $C00F cuando el intérprete del BASIC se dirija a la $73, entrando así en acción nuestro programa.



La dirección $C00E se deja en vacío, mediante .BYTE 00, destinándola para almacenamiento temporal de los códigos de error que serán empleados más adelante.



LOCALIZACION DE LOS COMANDOS GRAFICOS

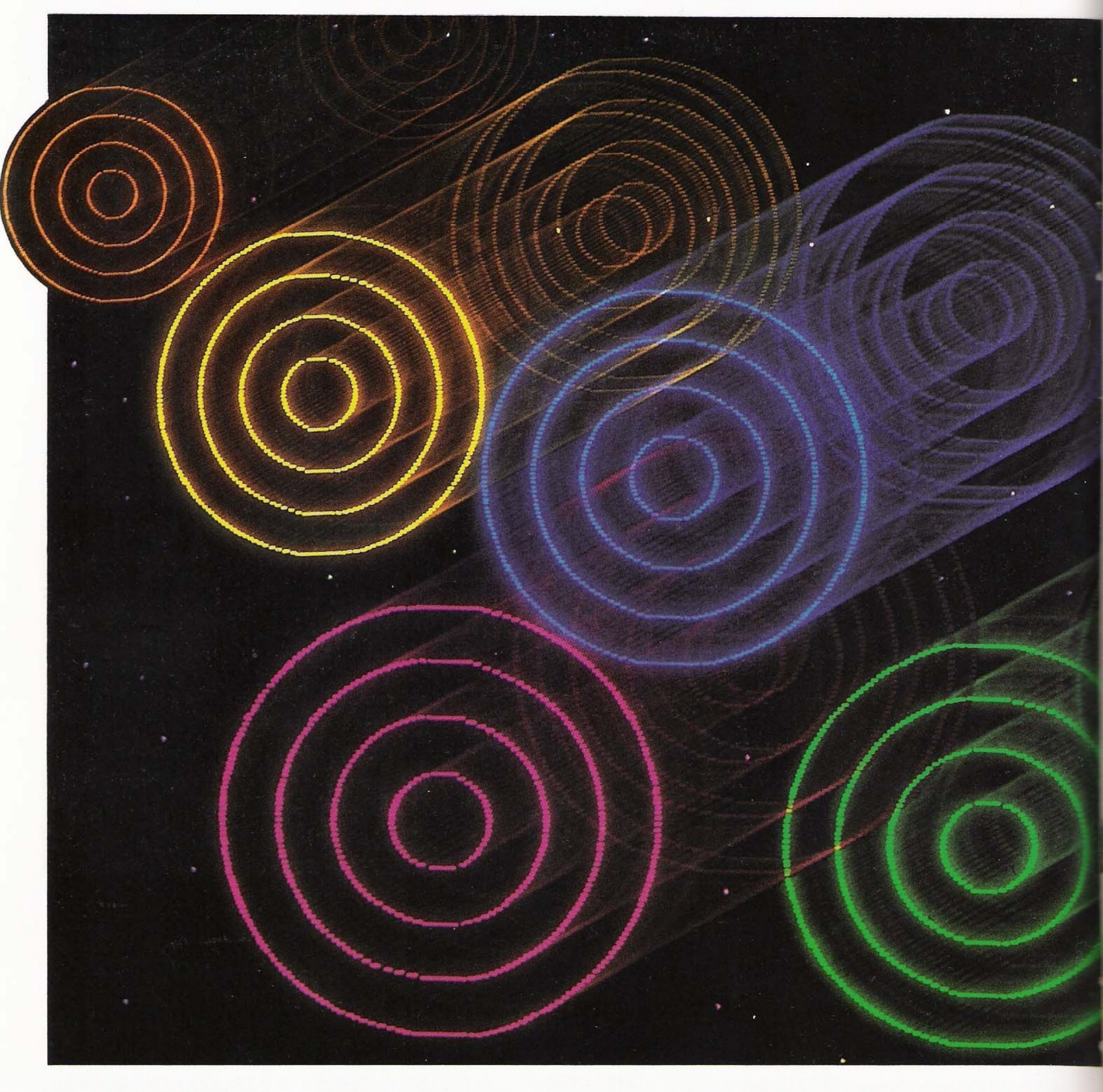
La rutina que viene a continuación busca la existencia de @. Las cuatro primeras instrucciones desplazan el puntero que señala a un byte dado del texto del programa en BASIC para que apunte hacia el siguiente byte. Este era un trabajo normalmente acometido por CHRGET, pero ahora lo efectúa nuestro programa.

```
*=$C00F;ORG 49167
        INC $7A
        BNE ON
        INC $7B
ON      STX $C00E
        TSX
        SEC
        LDA $9D
        BEQ RUN
        LDX #$0D
        JMP $0079
RUN     JSR $0079
        CMP #$40
        BEQ $C030
        LDX $C00E
        JMP $0079
```

```
:C00F E6 7A D0 02 E6 7B 8E 0E
:C017 C0 BA 38 A5 9D F0 05 A2
:C01F 0D 4C 79 00 20 79 00 C9
:C027 40 F0 06 AE 0E C0 4C 79
:C02F 00
```

```
10 T=0
20 FOR I=49167 TO 49199
30 READ A
40 POKE I,A
50 T=T+A
60 NEXT I
70 IF T<>3685 THEN PRINT
   "ERROR EN LOS DATA"
100 DATA 230,122,208,2,230,
    123,142,14,192,186
110 DATA 56,165,157,240,5,
    162,13,76,121,0
120 DATA 32,121,0,201,64,
    240,6,174,14,192
130 DATA 76,121,0
```

LDA $9D y BEQ RUN comprueban si está siendo ejecutado un pro-



grama en BASIC. De ser así, el valor de la variable del sistema almacenado en la dirección $9D será 0.

Por el contrario, si no está corriendo algún programa, el registro X será cargado con el byte $0D, que es el código correspondiente a *sintax error*. JMP $0079 devuelve el control a una parte de la rutina CHRGET, bautizada como CHRGOT, que continúa imprimiendo el mensaje de error en la pantalla. Esta rutina no trabajará en modo directo, sólo en el caso de programa.

Si está ejecutado un programa escrito en BASIC, el procesador saltará hasta la instrucción JSR $0079, que es la subrutina CHRGOT y toma de nuevo el mismo byte del programa. La diferencia con el salto anterior (JMP) estriba en que JSR significa «salta a la subrutina que comienza en la dirección $0079», de tal manera que cuando encuentre una instrucción RTS (Retorna de la Subrutina) volverá a nuestro programa, en lugar de continuar hacia la rutina de impresión



en pantalla, contenida en la ROM. El byte del programa en BASIC es ahora comparado con #$40, equivalente a la @ en código ASCII. Si la comparación arroja un resultado positivo, el procesador pasa a la siguiente rutina. Si no, se carga en el registro X el byte de error presente en la dirección $C00E y salta de nuevo a CHRGOT, para visualizar un mensaje de error o continuar leyendo el siguiente byte.

ALMACENAMIENTO DE LOS PUNTEROS

Esta sección se encarga de los punteros.

```
*=$C030;ORG 49200
        JSR $0073
        LDX $7A
        STX $FB
        LDX $7B
        STX $FC
        LDX #$01
        STX $FE

:C030 20 73 00 A6 7A 86 FB A6
:C038 7B 86 FC A2 01 86 FE

 10 T=0
 20 FOR I=49200 TO 49214
 30 READ A
 40 POKE I,A
 50 T=T+A
 60 NEXT I
 70 IF T<>2046 THEN PRINT
    "ERROR EN LOS DATA"
100 DATA 32,115,0,166,122,
    134,251,166, 123, 134
110 DATA 252,162,1,134,254
```

El siguiente byte del programa en BASIC es normalmente obtenido empleando CHRGET. El puntero que señala al byte actual es guardado en las direcciones de memoria $FB y FC. Esto se hace para que puedan ser manipuladas sin destruir a las variables del sietema.

En la dirección $FE se almacena un 1, siendo empleada como contador para la **palabra clave** destinada a los comandos gráficos con la que estamos trabajando en un momento dado.

En el próximo número incorporaremos los comandos HIRES y COLOUR (Alta Resolución y Color) a nuestro programa. Después continuaremos con MULTI, NRM, CIRCLE, DRAW, FLASH, etc.

Mientras esperas, puesdes ir tecleando la siguiente rutina en culaquiera de sus tres modalidades.

```
*=$C03F;ORG 49215
       LDX #$00
       CMP $CF00,X
       BEQ $C061
OTRVEZ LDA #$0D
       CMP $CF00,X
       BEQ STOP
       LDA #$01
       CMP $CF00,X
       BEQ $C0A4
       INX
       JMP OTRVEZ
STOP   INX
       INC $FE
       JSR $0079
       JMP $C041

:C03F A2 00 DD 00 CF F0 1B A9
:C047 0D DD 00 CF F0 0B A9 01
:C04F DD 00 CF F0 50 E8 4C 46
:C057 C0 E8 E6 FE 20 79 00 4C
:C05F 41 C0

 10 T=0
 20 FOR I=49215 TO 49248
 30 READ A
 40 POKE I,A
 50 T=T+A
 60 NEXT I
 70 IF T<>4408 THEN PRINT
    "ERROR EN LOS DATA"
100 DATA 162,0,221,0,207,
    240,27,169,13,221
110 DATA 0,207,240,11,169,
    1,221,0,207,240
120 DATA 80,232,76,70,192,
    232,230,254,32,121
130 DATA 0,76,65,192
```

Programación

NO LO OLVIDES, ARCHIVALO

- VENTAJAS DE ALMACENAR DATOS EN UN PERIFERICO
- LA MARCA FIN DE FICHERO
- TERMINADORES MUDOS
- USO DE LOS DATOS

Teclea para Commodore-64

Abre un canal hacia tu *cassette* o unidad de discos y envía por él tus datos importantes. Saber manejar ficheros con soltura es una de las claves de la programación.

Todo programa que maneje gran cantidad de datos ha de contar con alguna forma conveniente de almacenarlos. Existen cortos programas que permiten almacenar un listín telefónico sencillo en sentencias de tipo DATA. El programa busca e imprime un dato determinado, pero no tiene manera de actualizar la lista sin interrumpir el programa y alterar las líneas del mismo. Además, solamente se puede almacenar la información dentro del propio programa. De esta manera está uno encadenado a la información que contenga, lo cual no es muy satisfactorio.

Evidentemente, es mucho mejor almacenar los datos separadamente en una cinta o disco, en lugar de mantenerlos dentro del programa. De esta forma, el ordenador puede leer la información con la misma facilidad con que lee las líneas del programa, pero se pueden modificar los datos con mucha mayor facilidad, y, además, el usuario del programa no tiene que ser un experto en programación para modificar dichos datos.

En una cinta o en un disco los datos se almacenan en un fichero, y muchos de los programas de aplicaciones almacenan los datos de esta forma. Así, el fichero de *hobbies,* finanzas caseras, calendario y hoja electrónica, que iremos publicando, son algunos de los programas que han de poder manejar grandes cantidades de información. Si examinas atentamente estos programas, identificarás la parte que almacena y carga los datos. En todos los casos es una pequeña parte del programa, siendo mucho más lo que se dedica a asuntos tales como la disposición en memoria, mensajes al usuario *(prompts),* visualizaciones y cosas por el estilo.

Almaceando la información separadamente del «núcleo» del programa, tienes lo que se llama una memoria virtual: el programa básico se puede usar con una variedad de ficheros diferentes, cada uno con un número distinto de registros o campos, con una disposición adecuada al objeto de cada fichero.

El almacenamiento secuencial de datos es uno de los métodos preferidos. Su programación es muy sencilla y los principios en que se basa son fáciles de entender. Además, los datos se pueden convertir a un formato que realmente permite pasarlos de un programa a otro o, incluso, de un ordenador a otro.

Un fichero secuencial difiere de los ficheros de otra clase en que almacena los datos **en serie,** siendo cada dato separado del siguiente por un solo byte. Toda la información se recoge en la memoria antes de ser transferida a un fichero para su almacenamiento.

El primer paso para guardar un fichero secuencial es introducir la información en tu dispositivo de almacenamiento. Esto se hace con el comando OPEN. Los comandos de uso más corriente se describen en los manuales del usuario.

Este artículo explica con detalle la forma de crear tus propios ficheros de modo que puedas incorporarlos en tus programas.

CREACION DE UN FICHERO

El programa muestra cómo realizar una versión más sofisticada del directorio telefónico mencionado anteriormente. Pero no tendrás dificultad en cambiar los mensajes de pantalla y cadenas alfanuméricas para adaptarlos a otros tipos de información.

Las entradas admitidas son nombre, apellido y número. Como los números de teléfono contienen espacios y posiblemente también guiones, es mejor tratarlos como una matriz unidimensional. Con otras clases de datos tal vez prefieras tener algunas cadenas de matriz y algunas cadenas numéricas. No importa el tipo de variable que se almacena, o en qué orden, ya que se vuelven a leer en el mismo orden y con el mismo tipo de variable.

Se puede enviar información directamente a un fichero, pero es más corriente y más conveniente almacenarla primero en una cadena, almacenando posteriormente toda ella.

La primera parte del programa establece un sencillo bucle que te permitirá introducir los datos. Puedes meter todos los nombres y números que quieras, hasta el número dimensionado en la línea 10. Si quieres meter más de 50 tienes que cambiar la dimensión. Cuando termines, en vez de seguir metiendo un nombre y número, pulsa RETURN y la subrutina de entrada se detendrá.

La segunda mitad del programa, desde la línea 100 en adelante, escribe los datos en el fichero. Esta parte se explica después de cada programa.

```
 1Ø DIM A$(5Ø),B$(5Ø),T$(5Ø)
 3Ø N=N+1
 4Ø INPUT"■■NOMBRE ■";
    A$(N)
 5Ø INPUT"■APELLIDO■";B$(N)
 6Ø INPUT"■ TELEFONO ■";
    T$(N)
 7Ø IF A$(N)<>"" AND N<5Ø
    THEN 3Ø
 8Ø PRINT"■GUARDANDO LOS
    DATOS"
1ØØ OPEN 1,1,1,"FICHERO"
11Ø PRINT#1,N-1:Q$=CHR(34):
    C$=","
12Ø FOR Z=1 TO N-1
13Ø PRINT#1,Q$A$(Z)Q$C$Q$B$
```




```
    (Z)Q$C$Q$T$(Z)Q$
140 NEXT Z
150 CLOSE 1
160 PRINT"DATOS GUARDADOS"
170 END
```

Para ficheros en disco, cambia la línea 100 por:

```
100 OPEN 1,8,1,"FICHERO"
```

El fichero se abre especificando el número de fichero, número y dirección del dispositivo, seguido por el nombre del fichero entre comillas. En este programa sólo se utiliza un fichero, de modo que el número de fichero es 1. El número de dispositivo es 1 para la cinta y 8 para el disco. La dirección es 1, lo cual significa que se está *escribiendo* en el fichero.

PRINT # 1 significa "escribe la siguiente unidad de información en el fichero 1", de forma que la primera parte de la línea 110 almacena el va-

lor de N − 1, que se utilizará más tarde, cuando se lean los datos.

El resto de la línea 110 hace la variable Q$ igual a comillas (cuyo valor en código ASCII es 34) y C$ igual a una coma. Estos caracteres se utilizan en la línea 130, que escribe los nombres y números en el fichero. Se utilizan las comas para separar cada grupo de datos, pero puedes usar otros separadores, como los dos puntos, el punto y coma o el código de RETURN (CHR$(13)). Las comillas se usan para rodear cada matriz unidimensional de datos. Las comillas constituyen una protección para el caso en que los datos introducidos contengan cualquier signo de puntuación que pudiera ser interpretado por el programa como un separador.

Con la instrucción CLOSE 1 en la línea 150 se cierra el fichero.

LECTURA DE UN FICHERO

El programa para leer el contenido de un fichero, y trasladarlo a la memoria, es el inverso del programa de escritura. También se dedica una línea para presentar la información en pantalla a fin de que puedas comprobar que realmente han sido leídos los datos.

```
200 CLR:OPEN 1,1,0,"FICHERO"
210 INPUT#1,N:DIM A$(N),
    B$(N),T$(N)
220 FOR Z=1 TO N
230 INPUT#1,A$(Z),B$(Z),
    T$(Z)
235 PRINT A$(Z),B$(Z),T$(Z)
240 NEXT Z
250 CLOSE 1
```

Una vez más, para ficheros de disco, cambia la línea 200 por:

```
200 OPEN 1,8,0,"FICHERO"
```

Para usar este programa, rebobina la cinta hasta el principio del fichero, y a continuación pulsa RUN 200 para leer los datos; no ejecutes (RUN) el programa entero otra vez.

La instrucción CLR en la línea 200 borra todos los nombres y números, para que puedas estar seguro de que la información que se visualiza ha sido realmente leída en el fichero. El resto de la línea abre el fichero para leer en él, observa que el tercer número después del OPEN ha cambiado de 1 a 0.

La línea 210 introduce el primer dato. Se trata del número de elementos del conjunto y se usa para establecer el final del bucle de lectura de datos. Cuando se conoce N, las cadenas se dimensionan con el tamaño correcto. La línea 230 lee entonces cada elemento unidimensional del fichero y la siguiente línea los presenta en pantalla. Aquí también el fichero se cierra con CLOSE 1.

EL FINAL DEL FICHERO

En el último programa se utilizaba la variable N en el bucle que controla el procedimiento de lectura, pero no siempre es posible saber de antemano cuántos datos hay que leer. Una manera de obviar este inconveniente es utilizar una marca de fin de fichero. El Commodore utiliza la palabra clave STATUS (o ST).

Según esto, el último programa se puede modificar para que siga imprimiendo datos hasta que la variable ST sea igual a 64.

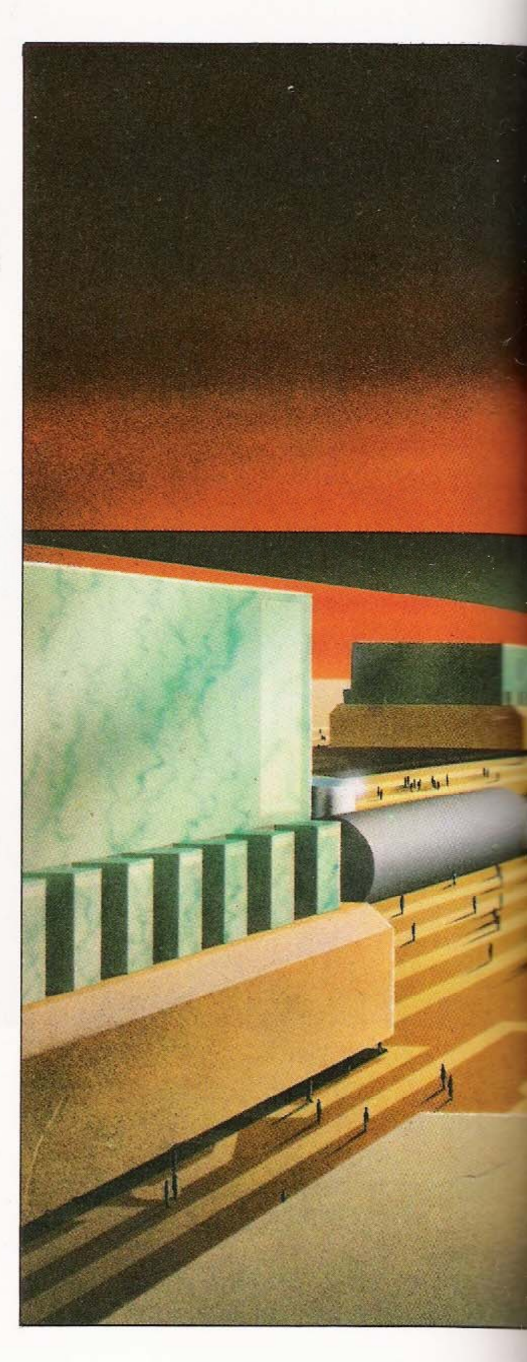
He aquí el programa completo a partir de la línea 100 para que puedas compararlo con la versión anterior. Guarda el programa viejo (con SAVE) y cambia las líneas que siguen.

Como siempre, si tienes una unidad de discos, cambia el segundo número después de OPEN por un 8.

En todos estos programas, la línea 240 comprueba si ha llegado al final de fichero, y aquí puedes ver la manera exacta de usar la marca de fin de fichero en tu ordenador. La línea 100 también se puede escribir de otra forma si se usa cinta. Prueba esto:

```
100 OPEN 1,1,1,"FICHERO"
105 Q$=CHR$(34):C$=","
110 Z=0
120 Z=Z+1
130 PRINT#1,Q$A$(Z)Q$C$Q$B$
    (Z)Q$C$Q$T$(Z)Q$
140 IF A$(Z)<>"" AND Z<50
    THEN 120
150 CLOSE 1
160 PRINT"DATOS GUARDADOS"
170 END
200 CLR:OPEN
    1,1,0,"FICHERO"
210 Z=0:DIM A$(50),
    B$(50),T$(50)
220 Z=Z+1
230 INPUT#1,A$(Z),B$(Z),
    T$(Z)
235 PRINT A$(Z),B$(Z),T$(Z)
240 IF ST<>64 THEN 220
250 CLOSE 1
```

```
100 OPEN 1,1,2,"FICHERO"
```



El número 2 del comando OPEN significa: "escribe una marca de fin de fichero cuando se cierra el fichero", y hace que la cinta se pare cuando llega el final del mismo.

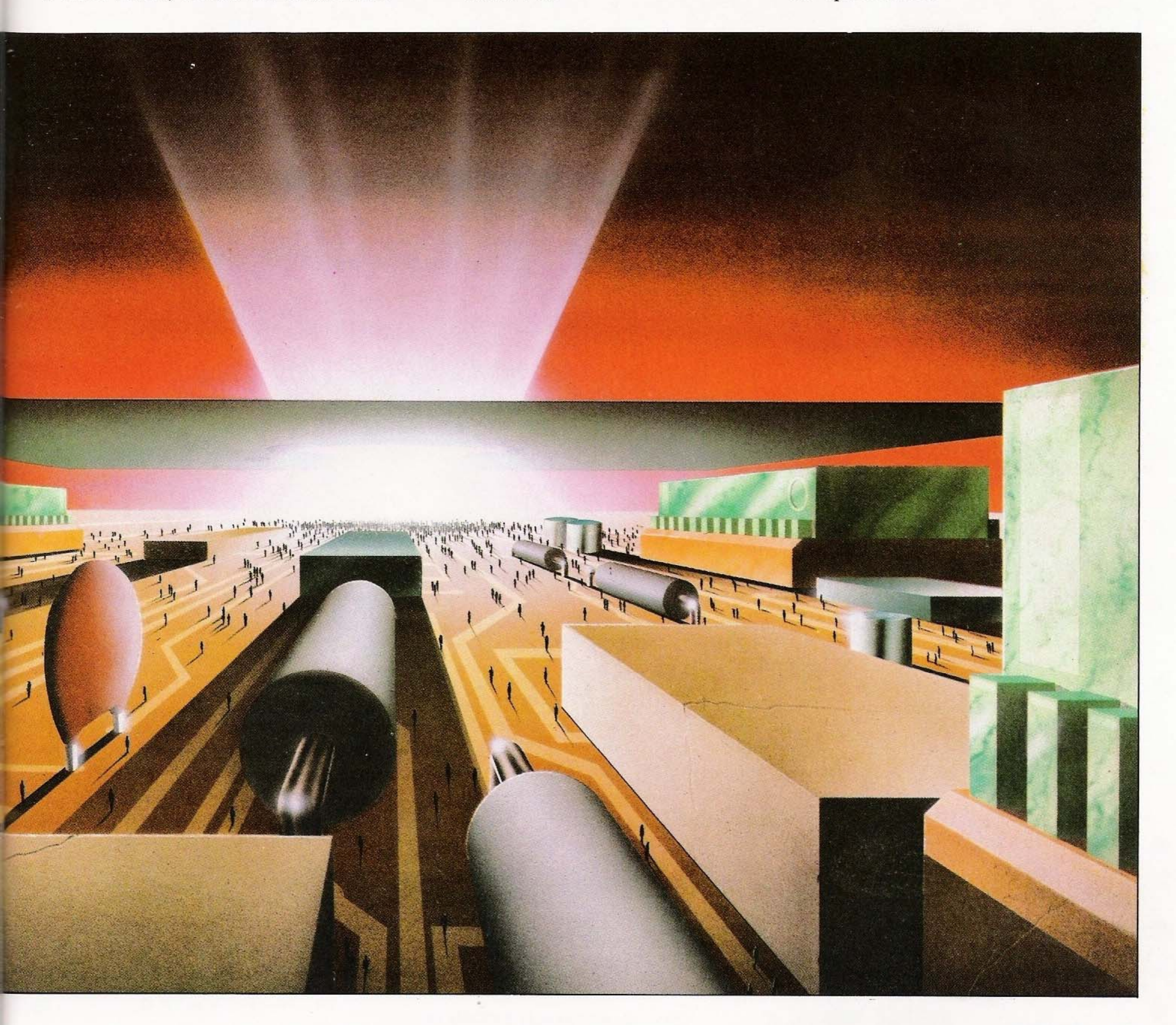
TERMINADORES MUDOS

Otra manera de detectar la llegada del final de un fichero es tener algún tipo de variable "muda". El procedimiento es muy simple. En la rutina original de entrada, cuando se empiezan a meter los datos en el ordenador, se puede introducir una palabra o valor mudo, detrás del último dato correcto. Podría ser la palabra FIN, o ZZZ o 999 o cualquier otra cosa que normalmente no vayas a encontrarte en el fichero real. Para ver cómo se utiliza en la práctica la variable muda, modifica las siguientes líneas. Cuando termines de meter los datos esta vez, teclea FIN, en lugar del siguiente NOMBRE, después RETURN para los otros dos datos pedidos en la pantalla.

```
70 IF A$(N)<>"FIN" AND N<50
   THEN 30
140 IF A$(Z)<>"FIN" AND Z<50
   THEN 120
240 IF A$(Z)<>"FIN" AND Z<50
   THEN 220
```

USO DE LOS DATOS

El fichero no es más que una forma de guardar los datos separadamente del programa. Tiene que ser leído por cualquier programa que haga uso del mismo. No obstante, todas las rutinas que se refieren al manejo de cadenas, en vez de ficheros, y las técnicas para su manejo se irán describiendo en sucesivos artículos de **INPUT** con distintas aplicaciones.



COMANDOS PARA CONTROL DE FICHEROS

La capacidad de manejo de ficheros del **Commodore 64** es mucho mayor que la de otros ordenadores, y la función concreta que desarrolla depende de la forma exacta del comando OPEN que se emplee. Conviene echar una ojeada a este tema antes de examinar los programas de este artículo. La forma general de sintaxis es:

OPEN nlf, nd, ds, "nombre, tipo, modo".

Las letras nlf significan **número lógico de fichero,** nd es **número de dispositivo,** ds, **dirección secundaria.** El nombre del fichero va seguido del tipo y del modo de uso del canal que se ha abierto.

Para abrir (OPEN) un fichero secuencial, lo primero que hay que decidir es el nlf. Para programas de aplicaciones que funcionen "solos", normalmente no hay necesidad de abrir más que un canal a la vez y se puede elegir un nlf más o menos arbitrario. Puede variar entre 1 y 255 y se puede seleccionar cualquier número dentro de este margen, aunque es una buena práctica mantenerse por debajo de 128, ya que los valores superiores están destinados a ser usados de otra forma.

El nlf se utiliza tanto en lectura como en escritura.

Los dispositivos que se pueden especificar son:

- 10. Teclado.
- 1. Unidad de cinta C2N.
- 2. *Port* de usuario RS232 (modem, impresora, etc.).
- 3. Pantalla (monitor o televisor).
- 4/5. Impresora (IEEE, serie).
- 6. *Plotter.*
- 8/9. Unidades de disco (1541).
- 10. 255 definido por el usuario.

Para el manejo de ficheros se utilizan normalmente los números de dispositivos 1, 8 y 9. Si se utiliza una unidad de disco, se debe emplear uno de los dos últimos números en la estructura del comando.

La dirección secundaria, sd, puede tomar un valor entre 2 y 14 (15 se reserva para comandos y errores específicos del disco) e indica el canal por el que se va a efectuar la entrada o salida.

Al utilizar un programa de manejo de ficheros basados en disco, se pueden abrir con OPEN hasta un máximo de tres ficheros secuenciales a la vez. Pero en tal caso se han de utilizar distintos valores de ds, ya que de lo contrario el primer fichero se cerraría (con CLOSE) automáticamente cuando se abriera otro fichero con la misma ds.

Los valores utilizados normalmente son:

- 0. Leer un fichero.
- 1. Escribir en un fichero.
- 2. Escribir en un fichero con marca de final de fichero.

Si no se especifica ds, el valor adoptado por defecto es 0.

A continuación viene entre comillas el nombre del fichero (que puede tener una longitud de hasta dieciséis caracteres), el tipo de fichero y el modo.

Es conveniente utilizar siempre nombres de ficheros, aunque no es una exigencia estricta para almacenar datos en cinta (sí lo es, en cambio, para almacenarlos en disco). Hay cinco tipos diferentes de ficheros para elegir. El fichero programa es el más común y es el valor por defecto si no te quieres molestar en especificar un tipo:

- S. Fichero secuencial.
- U. Fichero de usuario *.
- P. Fichero de programa.
- R. Fichero relativo*.
- D. Destruir*.

(* indica que se usa sólo con disco.)

El último parámetro del comando, **modo,** indica al ordenador cómo utilizar el canal que se acaba de abrir, por medio de una de las siguientes letras:

- W. Escribir un fichero.
- R. Leer un fichero.
- A. Añadir a un fichero secuencial.
- M. Leer un fichero sin cerrar.

Todos los parámetros del comando, excepto el primero, se pueden omitir, adoptándose por defecto el canal del *cassette,* dispositivo 1.

Así pues, tienes muchas formas de emplear la instrucción OPEN. Aquí tienes algunos ejemplos sencillos:

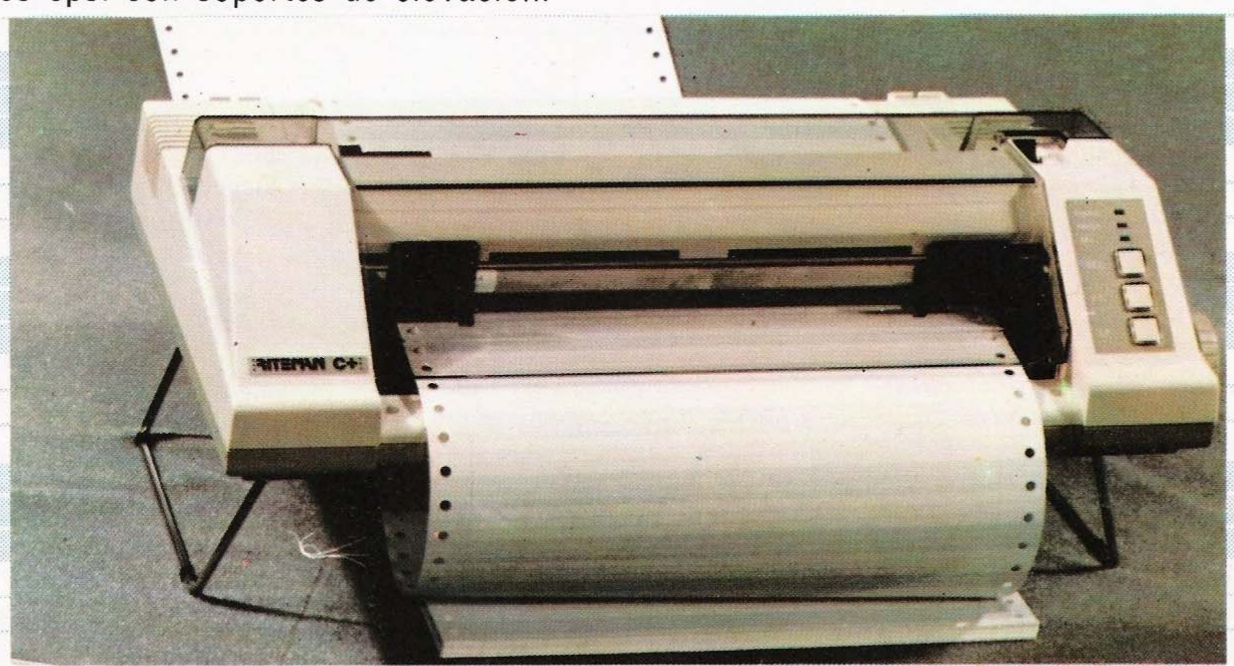
OPEN Z,Z,Z, "FICHERO"
escribe en cinta
OPEN 1,1,0, "FICHERO"
lee de la cinta
OPEN 1,8,1, "FICHERO"
escribe en el disco
OPEN 1,8,0, "FICHERO"
lee del disco
OPEN 1,1,1, "FICHERO,S,W"
escribe en cinta un fichero secuencial
OPEN 1,1,1, "FICHERO,S,R,"
lee de la cinta un fichero secuencial
OPEN 1,8,1, "FICHERO,S,W"
escribe en disco un fichero secuencial

:RITEMAN:

PERSONAL/BUSINESS PRINTER

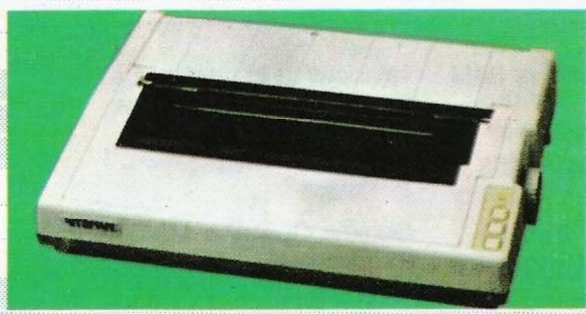
AMPLIA GAMA

Nuevas impresoras modelos **F+** y **C+**, sin rodillo alimentación horizontal, impresión vertical, tracción y fricción desde 4 a 10", bidireccional optimizada velocidad 105 cps. con soportes de elevación.

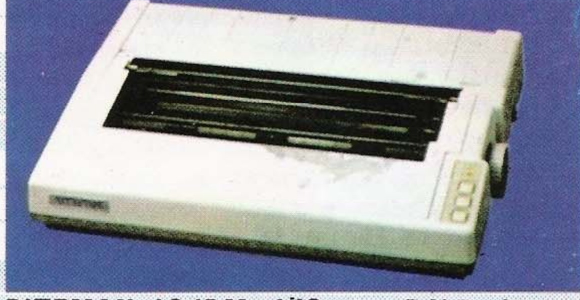


RITEMAN F+: Interface Paralelo Centronics, 2K buffer NLQ **P.V.P. 69.000 pts.**
RITEMAN C+: Especial directa a COMMODORE (cable inc.) P.V.P. 67.000 pts.

Otros modelos RITEMAN en 80 y 136 columnas, velocidad 120, 140, 160 cps.



RITEMAN 10, 120 cps. P.V.P. 81.000



RITEMAN 10-IBM, 140 cps. P.V.P. 85.000



RITEMAN 10-II 160 cps. P.V.P. 93.000

RITEMAN 15 160 cps. P.V.P. 155.000

DE VENTA EN LOS MEJORES ESTABLECIMIENTOS ESPECIALIZADOS



DATAMON, S. A.

PROVENZA, 385-387, 6.º, 1.ª
TELÉFONO (93) 207 27 04*

REPRESENTACION EN ESPAÑA DE:

:RITEMAN:

-IMPRESORAS PROFESIONALES-

08025 - BARCELONA

*** MAYORES PRESTACIONES**
*** MENOR TAMAÑO**
*** MEJOR PRECIO**

QUE ES LA HOJA DE TRABAJO

- QUE ES LA HOJA DE TRABAJO
- CALCULOS CON ORDENADOR
- PREDICIENDO EL FUTURO
- LA HOJA DE TRABAJO TIPICA
- PLANIFICACION Y DISEÑO

Si tienes que manejar muchas cifras, es un buen momento para hacer que te ayude tu micro. Probablemente es mucho mejor que tú con los números.

Una penosa tarea, que casi todo el mundo se ve obligado a realizar, es el control de los gastos propios, lo que constituye una buena idea de aplicación para el usuario de un micro, que puede servirse del mismo para saber a dónde va a parar su dinero. Pero ahora nos vamos a referir a un sistema distinto, modelado a partir de otro que utilizan los contables profesionales: la hoja de trabajo (también llamada hoja electrónica).

Las hojas de trabajo figuran entre los programas más versátiles, con unas posibilidades casi ilimitadas para el manejo de información numérica. Además, no están en absoluto restringidos al uso de datos financieros.

Este artículo consta de tres partes. Para empezar daremos una visión de lo que puede hacer una hoja de trabajo y para qué se utiliza. Seguidamente podrás programar tu propia y sencilla hoja, utilizando el listado que empieza en la segunda parte de este artículo. En la última parte te daremos instrucciones detalladas que harán por ti una parte del trabajo.

QUE ES UNA HOJA DE TRABAJO

Las hojas de trabajo utilizan una de las mayores ventajas que tienen los ordenadores: su capacidad de hacer cálculos muy rápidamente. En esencia hasta el ordenador más grande no es más que una máquina de sumar. De hecho, lo único que maneja un ordenador son números, y quienes se han sumergido en el código máquina refrendarán esta afirmación.

La hoja de trabajo puede ser una herramienta enormemente potente. Se utiliza normalmente para realizar cálculos financieros, pero también puede servir para construir toda clase de modelos de cálculo. Sustituye a los viejos métodos a base de lápiz, papel y calculadora, utilizados por los contables para calcular los beneficios de una empresa. A nivel doméstico la hoja de trabajo puede servir para el seguimiento de los gastos personales, o para tratar los datos relacionados con un *hobby*.

La hoja de datos usada tradicionalmente en contabilidad, en la que por ejemplo se registran los ingresos y los gastos, consiste en una gran hoja de papel, que normalmente ocupa la extensión de un par de páginas. Se divide horizontalmente en filas y verticalmente en columnas. Se produce así una retícula de cajas, celdillas o «células». En la fila superior el contable escribe, por ejemplo, los nombres de los meses del año, de forma que cada columna se refiere a un mes. Junto a la cuadrícula se disponen encabezamientos, tales como ingresos y gastos. Para un análisis más detallado puede introducir subtítulos tales como ventas propias, exportaciones, costes laborales, costes materiales brutos, etc. De este modo cada fila está relacionada con un área específica de ingresos y gastos.

El título final en el margen de la cuadrícula es normalmente pérdidas/ganancias, y las cifras que figuran al final de cada columna muestran el beneficio o pérdida habidos cada mes. Al final de cada fila, en la columna número 13, figura el ingreso o gasto total en cada área espcífica al cabo del año.

Rellenar las celdillas con los números es una tarea laboriosa, se disponga o no de un ordenador. Pero los contables que utilizan una hoja de trabajo de papel, han de abordar además la pesada tarea de calcular las cifras de pérdidas o ganancias. Esto supone la suma de todas las cifras de ingresos, la de todas las cifras de gastos y posteriormente restar el total de gastos del total de ingresos.

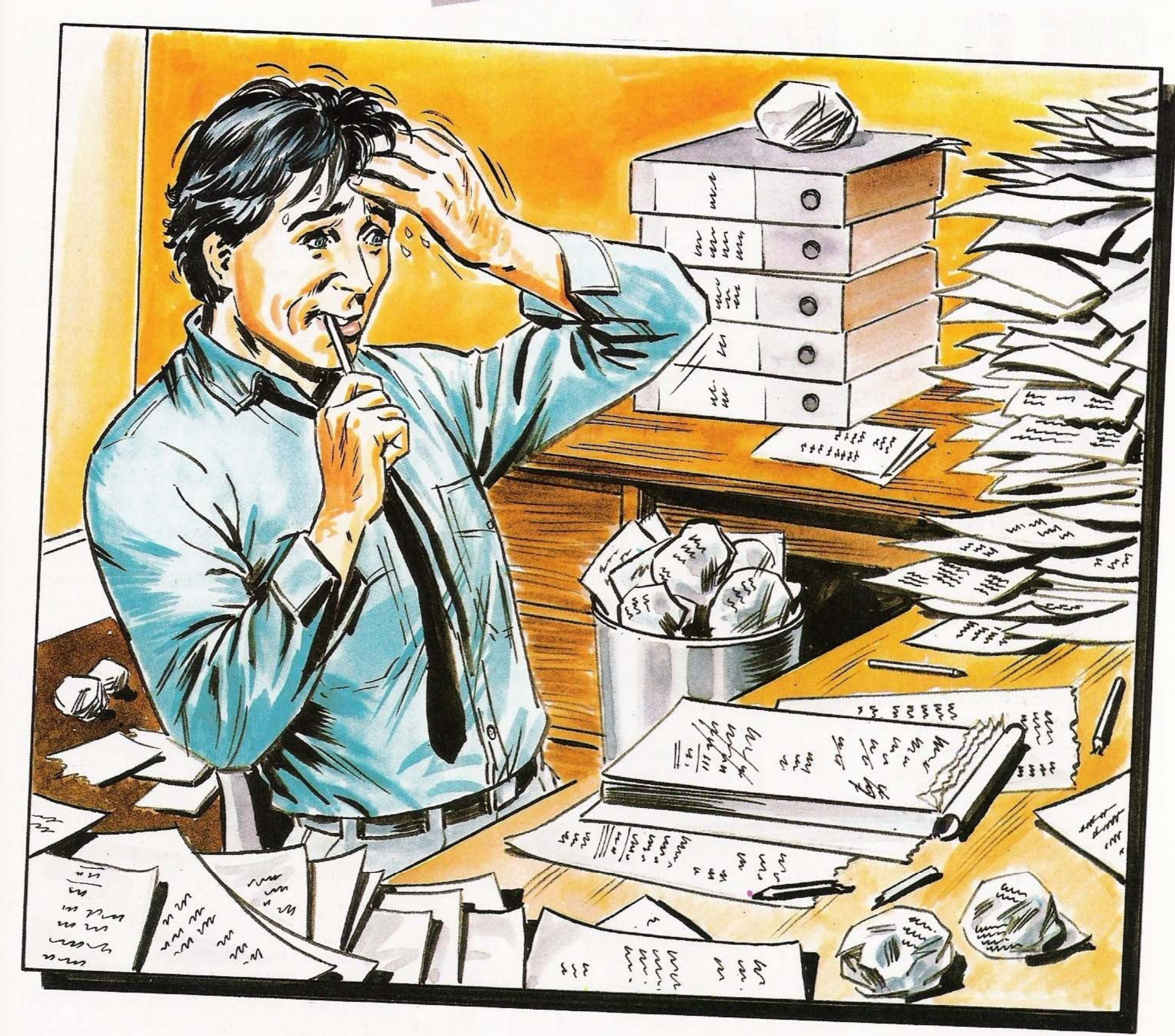
ENTRA EL ORDENADOR

En muchos aspectos, la hoja de trabajo de un ordenador es igual que la que utiliza papel, con la misma cuadrícula, dividida en filas y columnas. En la práctica, para que las casillas tengan un tamaño razonable, sólo se representa en pantalla una pequeña sección de la hoja; de esta manera la pantalla actúa como una «ventana» que permite observar la zona particular en que estés interesado.

Aquí también, igual que en la versión de papel, puedes meter lo que quieras en las casillas vacías, que no tienen significado alguno hasta que no las defınas. Puedes introducir una etiqueta o título, o números, dependiendo de lo que quieras que presente la hoja de trabajo.

Hasta aquí, la hoja en el ordenador es, si acaso, un poco más complicada que una hoja de papel. Pero su verdadera potencia reside en la capacidad de procesar la información que introduces en ella. Por debajo de la hoja en blanco, que tú vas rellenando, existe otra hoja escondida. Esta segunda hoja de trabajo es la que dice al ordenador lo que debe ir haciendo con la información que encuentra en cada casilla. Sin embargo, dicha hoja de trabajo «escondida» no tiene nada de misterioso, ya que eres tú quien la pones allí, y siempre está disponible para que puedas examinarla o modificarla en cualquier momento.

Volviendo a nuestro esforzado contable, supongamos que quiere representar en una segunda columna el tanto por ciento de impuestos corres-



pondiente a determinado precio, y en otra columna la suma de las dos cantidades anteriores. Con el ordenador, puede hacer un programa que haga todo esto a su gusto. Todo lo que tiene que hacer es introducir una instrucción en cada una de las celdillas de la columna dos, que mande al ordenador multiplicar el número de la columna uno por un porcentaje fijo. Una instrucción análoga en cada una de las casillas de la columna tres, hará que el ordenador calcule el total requerido, sumando los contenidos de las correspondientes casillas de las dos columnas anteriores.

PREDICIENDO EL FUTURO

Otro problema con el que se enfrenta el contable con su gran hoja de papel, es la introducción de modificaciones. Por ejemplo, un aumento de los costes laborales supone volver a calcular todas las cifras de gastos y restarlas otra vez de las cifras de ingresos, para encontrar el valor actualizado de pérdidas y ganancias. Si solamente estás anotando cifras, la cosa puede no ser muy grave, pero si estás haciendo un pronóstico a un año vista, o aún más lejos, podría significar centenares de nuevos cálculos. Esta es la típica tarea que, aunque se realice con ayuda de una calculadora, requiere mucho tiempo, aburre a cualquiera y está muy sujeta a errores. Una labor de este tipo la hace un ordenador en unos milisegundos.

Si introduces correctamente los números en la hoja de trabajo —cosa no siempre tan sencilla como pudiera parecer—, cualquier cambio en una cifra producirá automáticamente los ajustes adecuados en todas las cifras relacionadas con ella.

Si, por ejemplo, se cambia la cifra

correspondiente a la fila del coste material bruto, el coste total quedará adecuadamente ajustado y se producirán los cambios necesarios en el beneficio total. Este es uno de los ejemplos más sencillos, pero algunas hojas de trabajo son capaces de realizar cálculos enormemente complejos. Esto las hace muy útiles para responder a preguntas del tipo «¿qué sucedería si...?», que constantemente necesitan una respuesta en los negocios, y en muchas otras áreas. Aunque la hoja de trabajo se utiliza principalmente para fines comerciales, también puede servir, por ejemplo, para predecir los cambios de población. De hecho cualquier situación en la que hay muchos valores variables independientes, constituye una potencial aplicación para una hoja de trabajo.

La potencia y versatilidad de las hojas de trabajo está haciendo que se convierta en uno de los tipos de *software* que mejor se venden. Muchas hojas de trabajo son compatibles con otro *software*, de forma que se puede constituir una biblioteca completa de *software* para aplicaciones más serias. Si se juntan un procesador de textos, un sistema de gestión de base de datos y una hoja de trabajo, resulta un «paquete» de programas enormemente potente, como por ejemplo el «Simphony», diseñado para el PC de IBM y compatibles.

UNA HOJA DE TRABAJO TIPICA

La unidad básica de la hoja de trabajo es la celdilla. El contenido de cada una puede ser una matriz unidimensional —por ejemplo una palabra— un número, o una fórmula. Cuando la hoja de trabajo se carga en el ordenador, las celdillas tienen una cierta anchura preseleccionada. En algunas hojas de trabajo, este valor atribuido «por defecto» se puede cambiar a voluntad, ya sea al comienzo de la sesión o en un momento posterior.

El valor representado en cada casilla puede ser un número que tú introduces o el resultado de un cálculo. El número de filas y columnas varía de una hoja de trabajo a otra, pero un valor corriente en las hojas de trabajo comerciales serias es 65 columnas y 256 filas. Esto supone 16.640 casillas individuales, lo cual es mucho trabajo para un micro.

El direccionamiento y localización de casillas se hace siempre por medio de letras o números a lo largo de los ejes X e Y de la cuadrícula, pero los detalles exactos de cómo se hace esto varían de una hoja de trabajo a otra. La mayoría utilizan una combinación de letras y números con las columnas designadas por A, B, C, ..., Z, siguiendo después con AA, AB, AC, ..., AZ, etc., para grandes hojas de trabajo. En tal caso las filas se designarían por números empezando por el 1. Este es el procedimiento seguido en el programa que sigue. La nomenclatura resulta entonces similar a la utilizada en los juegos de barquitos.

Hay varios comandos disponibles para introducir ecuaciones, valores o etiquetas, copiar celdillas o mirar diferentes partes de la hoja. Otros comandos sirven para realizar los cálculos y te permiten cargar y extraer los datos. Las ecuaciones pueden manejar las operaciones aritméticas usuales —sumar, restar, multiplicar y dividir— así como tantos por ciento y totales en cualquier fila y columna.

PLANIFICACION Y DISEÑO

La primera fase de la utilización de una hoja de trabajo es una de las más difíciles. Con frecuencia, requiere mucha planificación, pero durante ella no se utiliza el ordenador. Antes de empezar, tienes que decidir exactamente lo que quieres que haga el ordenador, ya que esto afectará a la manera de diseñar tu hoja de trabajo. Una hoja de trabajo correctamente concebida constituye un método ideal de presentación de información de una manera clara y concisa. Pero, como suele ocurrir con frecuencia en el cálculo, tu hoja de trabjo será lo buena que tú quieras hacerla. Si el trabajo se plantea de una forma descuidada, el resultado será una hoja de trabajo mal presentada, desordenada, difícil de leer, que más bien que revelar la información parece que la esconde.

Como ejemplo práctico, supongamos que quieres confeccionar una hoja de trabajo que te ayude a organizar las finanzas domésticas para todo el año. Evidentemente, tendrás

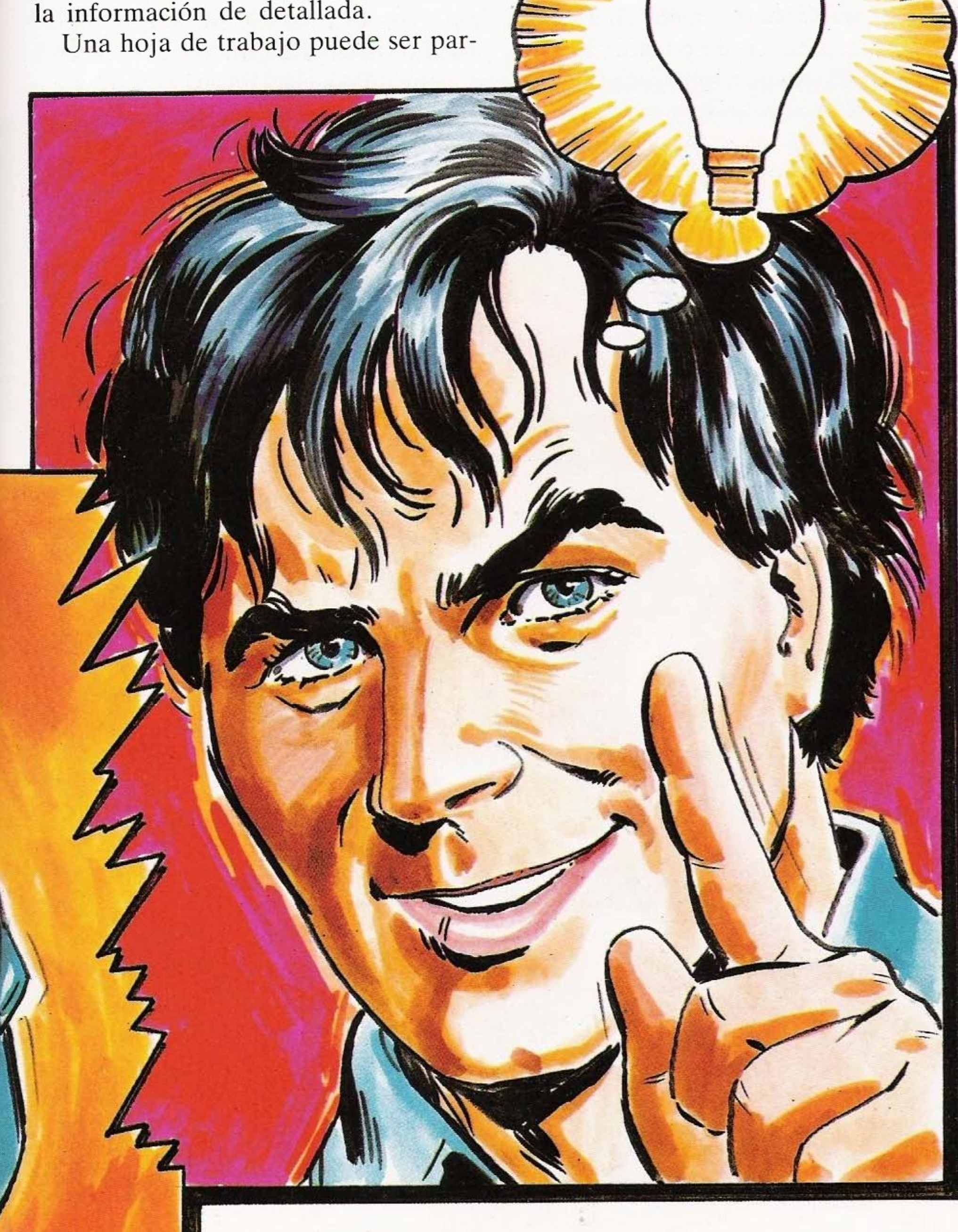


que utilizar los meses del año como encabezamiento de cada columna, en la parte superior de la hoja. Pero decidir el título de cada fila ya es algo más difícil.

Ante todo, ¿cuál es el grado de detalles que quieres? Algunas partidas obvias de la casa son: hipotecas, contribuciones, calefacción, seguros y mantenimiento. Pero los gastos de reforma y mejora, ¿quieres que se consideren como una categoría independiente? ¿O quieres incluir los costes del coche —gasolina, impuestos, seguro, mantenimiento, reparaciones— en una categoría conjunta, y llamarla algo así como gastos generales? Realmente todo depende de cómo quieras la información de detallada.

Una hoja de trabajo puede ser particularmente útil para el seguimiento del valor de tus inversiones, tales como la casa o el coche. Tendrías que poder averiguar en qué tanto por ciento se está revalorizando tu casa o se está depreciando tu coche.

El cálculo del incremento anual del valor de tu casa parece en principio muy sencillo. El valor correspondiente al año siguiente a la compra será el precio que pagaste por ella multiplicado por el tanto por ciento de crecimiento anual, es decir, P*X%, donde X es 100 más el tanto por ciento anual de incremento. Por ejemplo, para un incremento anual de 0,5%, X sería 100,5%. La fórmula para obtener el valor al cabo del segundo año sería P*X%*X%. La fórmula se hace aún más larga para el tercer año y para el final del décimo año es ya imposible de manejar.



Con una hoja de trabajo hay un procedimiento más sencillo. Afortunadamente no tienes que ser un matemático que domine docenas de complicadas fórmulas, para sacarle todo su jugo a la hoja de trabajo. En un caso como éste puedes utilizar la dirección de una celdilla para referirte al contenido de la misma. En este caso, la complicación de la fórmula no pasa de P*X%, donde P es el contenido de la celdilla anterior. En la fórmula que escribirías para la hoja de trabajo, P sería realmente la dirección de la celdilla anterior y podría ser algo como B10*100,5%.

Si introduces la fórmula en la celdilla C10 la respuesta aparecerá en dicha celdilla. Si introduces en la celdilla D10 la fórmula C10*100,5%, el ordenador toma el número representado en C10 y lo multiplica por 100,5%. La forma real que han de adoptar las ecuaciones varía de una hoja de trabajo a otra, y los programas que siguen utilizan un método algo diferente. Sin embargo los detalles concretos se explicarán en las instrucciones sobre cómo usar el programa, que veremos más adelante.

El uso de la dirección de cada celdilla en vez del contenido de la misma, hace que resulte muy fácil trabajar con una hoja de trabajo. Hace que casi cualquiera pueda efectuar cálculos matemáticos muy complicados con la ayuda de un poco de paciencia y sentido común.

¿QUE SUCEDERIA SI...?

Tu hoja de trabajo para el presupuesto doméstico te permitirá responder a toda clase de preguntas del tipo ¿qué sucedería si...? ¿Qué ocurriría si la hipoteca aumenta en junio un 2%? ¿Y si me compro un coche? ¿Y si instalamos calefacción central y/o dobles ventanas? Y así sucesivamente.

De hecho, este último ejemplo apunta hacia otra área donde se pueden utilizar las hojas de trabajo, aparte de las previsiones financieras y presupuestarias. Con una sola ojeada puedes ilustrar la diferencia entre sistemas de calefacción central que utilicen combustibles distintos. Y si puedes estimar la pérdida de calor que se evitaría con el uso de dobles ventanas, podrías encontrar lo que te ahorrarías y el tiempo que se tardaría en amortizar el coste de la instalación. La hoja de trabajo te puede valer incluso para dimensionar el tamaño correcto y el número de radiadores requeridos para calentar el piso o la casa a la temperatura adecuada.

OTROS POSIBLES USOS

Aunque hasta la más sencilla hoja de trabajo se puede utilizar en aplicaciones muy serias y complicadas, también pueden resultar divertidas. Igualmente, se pueden construir modelos distintos de los financieros usuales.

En el nivel más sencillo, y como puro entretenimiento, se puede crear una referencia circular por todas las casillas, que siga circulando indefinidamente.

Como ocurre con el *software* de utilidades muy difundido, de una hoja de trabajo a otra puede haber enormes variaciones de diseño y complejidad.

Como norma general, se puede decir que cuanto más potente es una hoja de trabajo, más caro resultará y más grande será el microordenador para el que esté diseñado.

TECLEA EL PROGRAMA

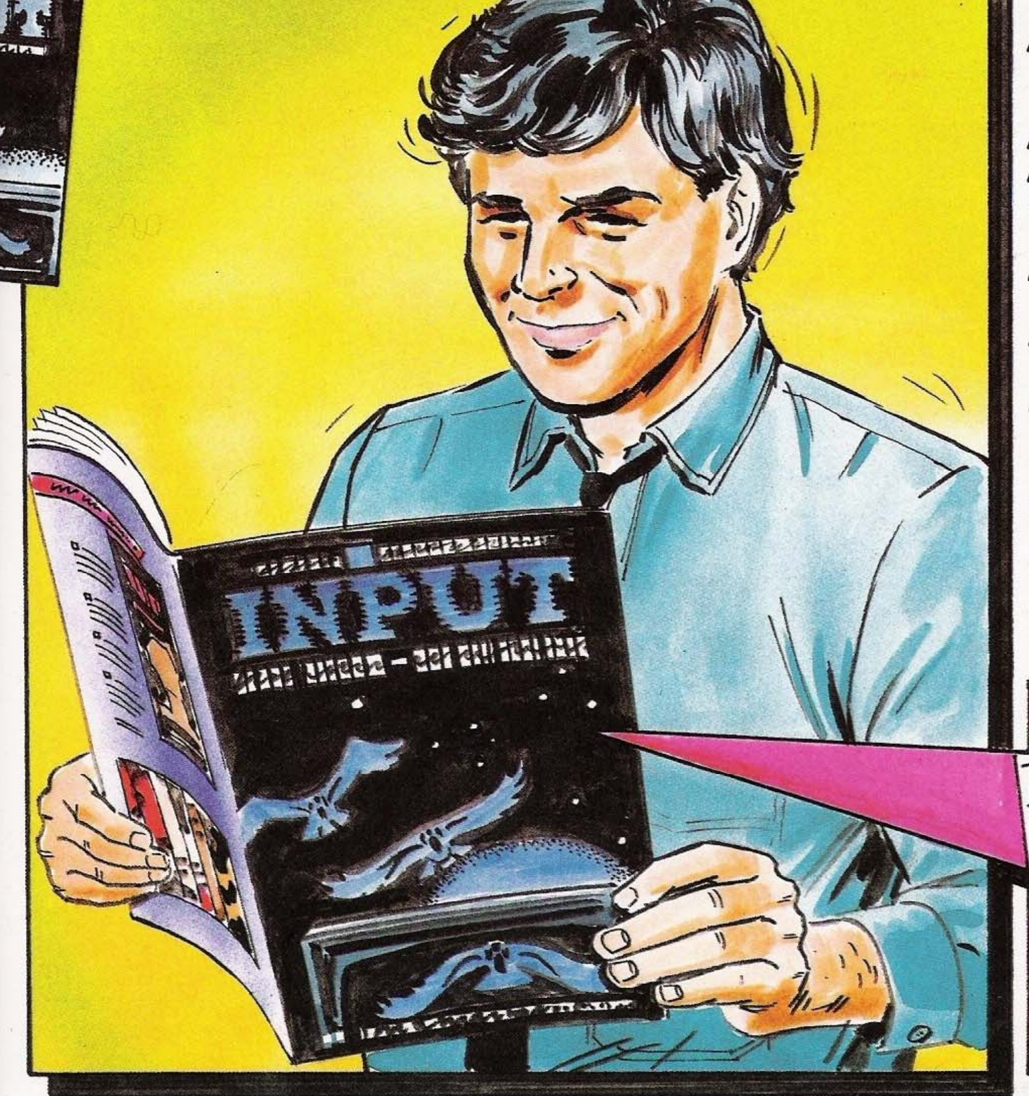
El programa de la hoja de trabajo es muy largo, por lo que lo publicaremos en tres partes. De momento, puedes ir copiando las líneas que damos a continuación y almacenándolas de forma que se puedan añadir más líneas posteriormente.

Teclea para Commodore-64

```
10 POKE 53280,4:POKE
   53281,0
20 PRINT"♥◥▀"SPC(16)
   "▜TRABAJANDO▀"CHR$(8)
30 RM=20:CM=24:LM=15
40 DIM D$(RM,CM)
50 A$=CHR$(128)+"□□□□□□□
   □□□□□□□□":AA$=A$:
   BB$=A$:F$=A$
60 FOR R=1 TO RM
70 FOR C=1 TO CM
```

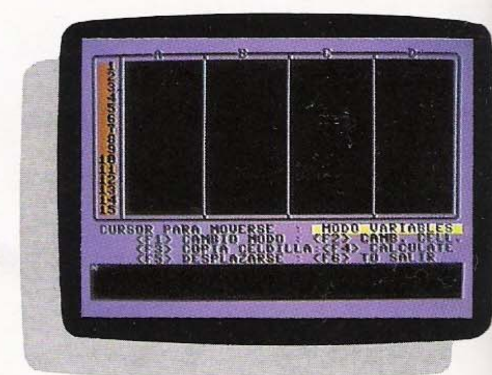


```
80 D$(R,C)=A$:NEXT C,R
90 GOSUB 1640
100 C$="■■■□"+CHR$(133)+
    CHR$(137)+CHR$(134)+
    CHR$(138)+CHR$(135)+
    CHR$(139)
110 DIM CL(3):CL(0)=5:
    CL(1)=30:CL(2)=158:
    CL(3)=159
120 OP$="+-*/%$&"
130 RS=1:CS=1:TP=0
140 GOSUB 210
150 IF Q<10 THEN 140
160 GOSUB 1560
170 PRINT"QUIERES SALIR DEL
    PROGRAMA (S/N)?"
180 AA$="S":BB$="N":GOSUB
    1280
190 IF A$="N" THEN 140
200 POKE 53280,14:POKE
    53281,6:PRINT"▣▣":END
210 FOR Z=0 TO 159:POKE
    1864+Z,32:NEXT Z
220 PRINT"■■■□⊟⊟";
230 FOR N=CS TO CS+3
240 PRINT"⊞⊟⊟⊟";CHR$(64+N)
    ;"⊟⊟⊟⊟";
250 NEXT N:PRINT"▯";
260 FOR R=RS TO RS+LM-1
270 AA$=STR$(R):IF
    LEN(AA$)<3 THEN
    AA$="□"+AA$
280 PRINT"■■▣"RIGHT$
    (AA$,2);:N=0
290 FOR C=CS TO CS+3
300 PRINT"■▯■"CHR$
    (CL(N));
310 IF TP=0 THEN PRINT
    RIGHT$(D$(R,C),8);
320 IF TP=8 THEN PRINT
    LEFT$(D$(R,C),8);
330 IF ASC(D$(R,C))=128 AND
    TP=8 THEN PRINT"□";
340 N=N+1:NEXT C
350 PRINT"■■▯";:NEXT R
360 POKE 198,0:PRINT"■□⊟⊟
    ⊟⊟⊟⊟⊟⊟⊟⊟⊟⊟⊟⊟⊟⊟⊟⊟⊟
    ⊟⊟⊟⊟⊟⊟⊟⊟⊟⊟⊟⊟⊟⊟⊟⊟⊟
    ⊟⊟□";
370 PRINT"■□CURSOR PARA
    MOVERSE□□:";
380 IF TP=8 THEN PRINT"□■
    □□MODO FORMULA□■□";Y.
390 IF TP=0 THEN
    PRINT"□π□MODO
    VARIABLES□■□";
400 PRINT"■□□□□<F1>
    CAMBIO
    MODO□:□<F2>□CAMB.□CELL
    .□";
410 PRINT"■□□□□<F3> COPIA
    CELDILLA:<F4>□CALCULATE
    □";
420 PRINT"■□□□□<F5>
    DESPLAZARSE□:□<F6>□TO
    SALIR□□□":PRINT"▣";
430 PRINT"*";:GET
    Q2$:PRINT CHR$(20);:IF
    Q2$="" THEN 430
440 Q1$=C$: GOSUB 2670
450 IF Q=0 THEN 430
460 IF Q=1 THEN CS=CS+1:IF
    CS>CM-3 THEN CS=CM-3
470 IF Q=2 THEN CS=CS-1:IF
    CS<1 THEN CS=1
480 IF Q=3 THEN RS=RS-1:IF
    RS<1 THEN RS=1
490 IF Q=4 THEN RS=RS+1:IF
    RS>RM-LM+1 THEN RS=RM-
    LM+1
500 IF Q=5 THEN GOSUB
    560:RETURN
510 IF Q=6 THEN GOSUB
    680:RETURN
```

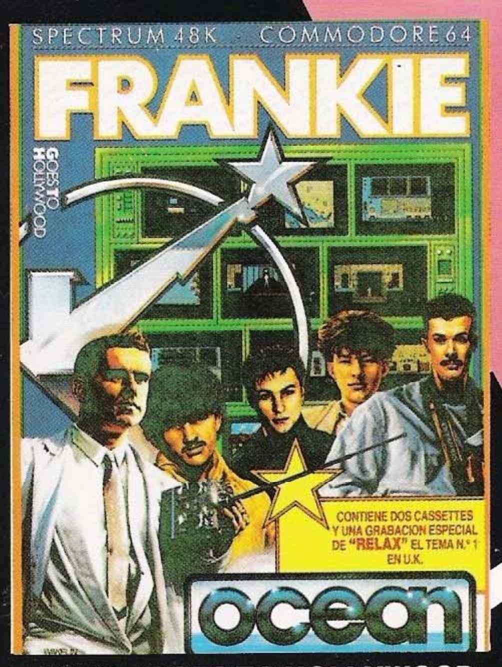


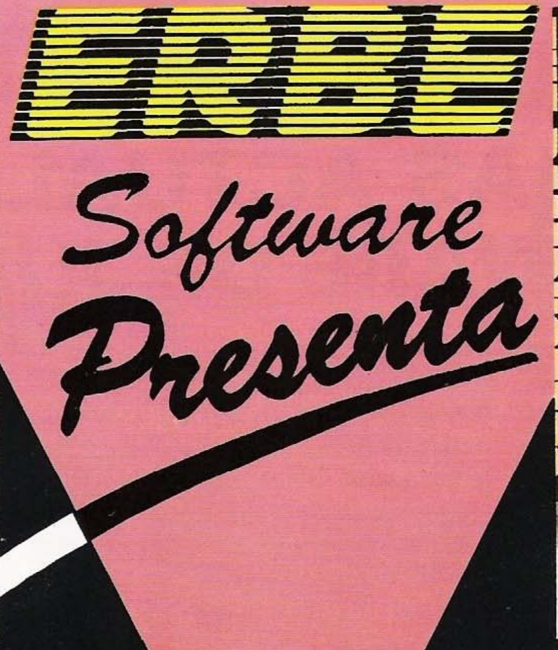


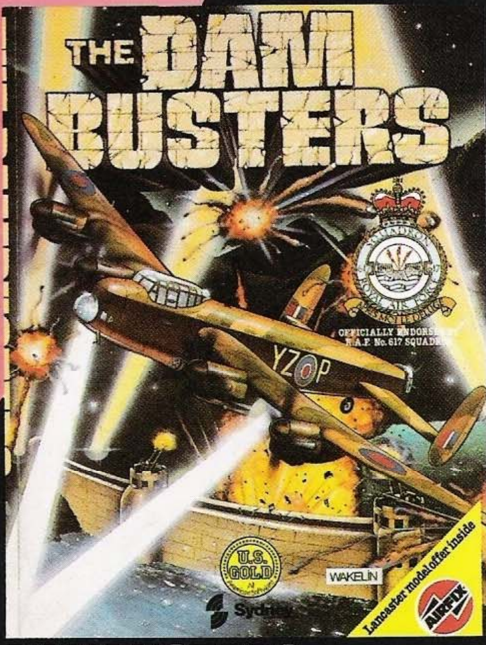
```
520 IF Q=7 THEN GOSUB
    1720:RETURN
530 IF Q=8 THEN GOSUB
    1080:RETURN
540 IF Q=9 THEN GOSUB
    2590:RETURN
550 RETURN
560 IF TP=0 THEN
    TP=8:RETURN
570 IF TP=8 THEN TP=0
580 RETURN
590 PRINT"QUE CELDILLA■?";
600 GET A$:IF A$=A$="" THEN
    600
610 IF A$="⇐"THEN RETURN
620 C=ASC(A$)-64
630 IF C<1 OR C>24 THEN
    600
640 PRINT"■"A$
650 INPUT R
660 IF R<1 OR R>20 THEN
    PRINT":□";:GOTO 650
670 RETURN
680 GOSUB 590:IF A$="⇐"THEN
    RETURN
690 PRINT"ENTRY■";
700 A$="":INPUT A$
710 IF TP=8 THEN GOSUB
    1040:GOSUB 750:
    D$(R,C)=A$+RIGHT$
    (D$(R,C),8)
720 IF TP=0 THEN GOSUB
    750:D$(R,C)=LEFT$
    (D$(R,C),8)+A$
730 IF LEFT$(D$(R,C),1)=
    CHR$(128) THEN D$(R,C)=
    "□"+RIGHT$(D$(R,C),15)
740 RETURN
750 IF LEN(A4)>8 THEN
    A$=LEFT$(A$,8)
760 IF TP=8 AND LEN(A$)<8
    THEN A$=A$+"□":GOTO
    760
770 IF TP=0 AND LEN(A$)<8
    THEN A$="□"+A$:
    GOTO 770
780 RETURN
790 AA$=MID$(A$,PS,3)
800 BB$=LEFT$(A$,1)
810 IF MBB$<"A" OR BB$>"X"
    THEN D1=0:RETURN
820 P=VAL(RIGHT$(AA$,2))
830 D1=2:IF P<10 THEN
    D1=1
840 IF P>CM OR P<1 THEN
    D1=0
850 RETURN
1640 RETURN
```



La pantalla muestra el diseño de nuestra hoja de trabajo. Como el programa es muy largo y en este artículo se ofrece solamente una parte, aún no es operativo. Sin embargo, ya puedes apreciar el característico diseño de las hojas de cálculo con las filas y las columnas que definen cada casilla.







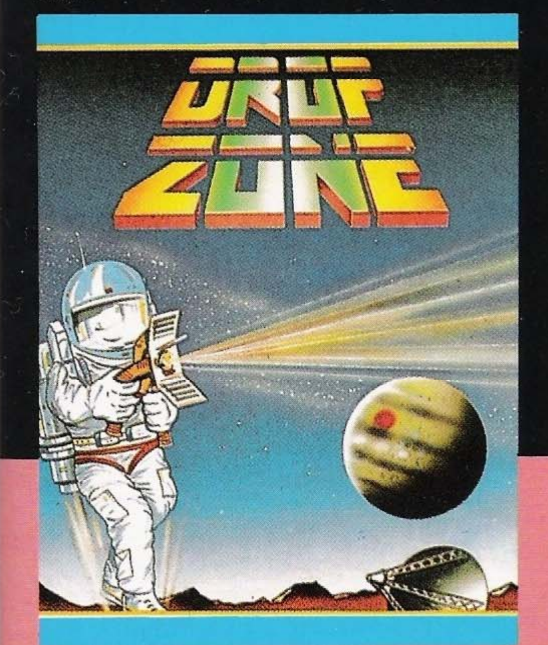
FRANKIE GOES TO HOLLYWOOD

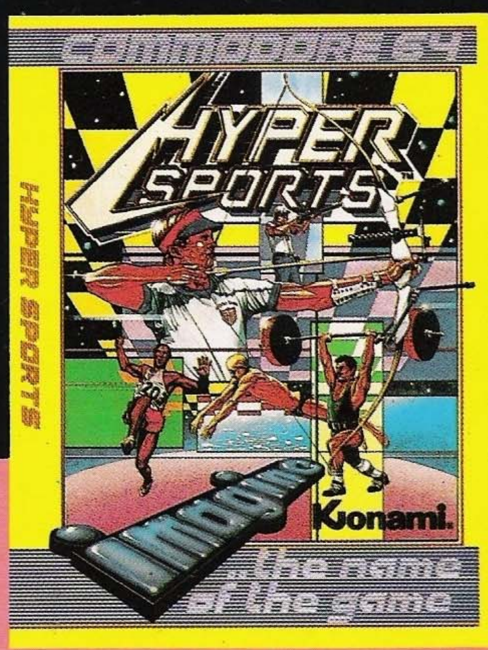
EL EXITO DEL MOMENTO. ALGO MAS QUE UN JUEGO. INCONTABLES PANTALLAS EN LAS QUE TODO PUEDE SUCEDER, DESDE TENER QUE DESCUBRIR AL ASESINO DE UN MISTERIOSO CRIMEN HASTA CONDUCIR UNA PELEA ENTRE REAGAN Y GORBACHOV. TU MISION ES SUPERAR 26 PRUEBAS DIFERENTES PARA PODER ENTRAR A LA SALA DEL PLACER. PRESENTACION DE LUJO QUE INCLUYE 2 CASSETTES Y EL TEMA N.º en U.K. "RELAX" EN GRABACION ESPECIAL.

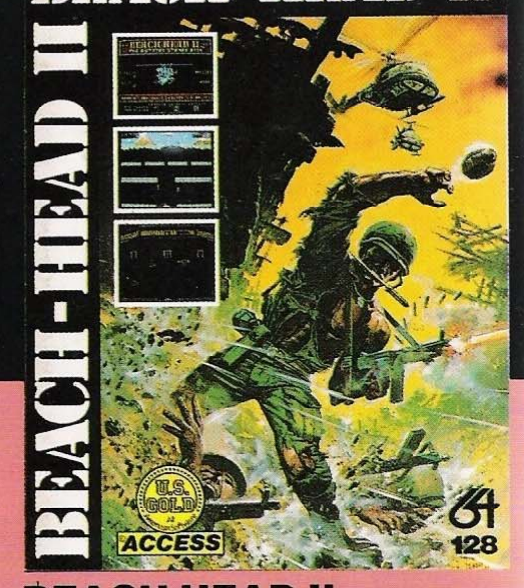
LO MEJOR PARA TU COMMODORE

DAMBUSTERS

OCUPA LOS PUESTOS DE PILOTO, INGENIERO DE VUELO, ARTILLERO, BOMBARDERO Y NAVEGANTE DEL LANCASTER QUE EN LA II GUERRA MUNDIAL DESTRUYO LAS PRESAS ALEMANAS, VOLARAS A TRAVES DEL CANAL DE LA MANCHA Y EUROPA ENFRENTANDOTE A LOS TERRIBLES ME-110 ALEMANES, ZEPPELINES, FOCOS ANTIAEREOS Y TODOS LOS DEMAS PELIGROS A LOS QUE SE ENFRENTO EL COMANDO INGLES.







DROPZONE

SENCILLAMENTE UN SUEÑO HECHO REALIDAD, POR FIN UN JUEGO QUE MANTENDRA A LOS AFICIONADOS A LOS PROGRAMAS DE ACCION ENTRETENIDOS DURANTE MESES. RAPIDEZ DE JUEGO TOTAL CONTROL, UNIDOS A EFECTOS SONOROS Y GRAFICOS FABULOSOS HACEN DE ESTE JUEGO DE CORTE ESPACIAL EL N.º DE SU CATEGORIA. ¡NO TE LO PIERDAS!

HYPERSPORTS

TIRO AL PLATO, NATACION, TRIPLE SALTO DE LONGITUD, LEVANTAMIENTO DE PESO, SALTO DE POTRO, Y TIRO CON ARCO SON LAS PRUEBAS QUE HAS DE SUPERAR EN ESTE FABULOSO JUEGO QUE ES EL ORIGINAL DE LAS MAQUINAS.

BEACH HEAD II

AQUI ESTA LA ESPERADA CONTINUACION DEL JUEGO MAS VENDIDO PARA EL C-64. BEACH HEAD II AÑADE A UNOS GRAFICOS Y EFECTOS SONOROS SOBERVIOS, UN SIMULADOR DE VOZ Y UN AUTENTICO COMBATE CUERPO A CUERPO QUE AUMENTA SU RELISMO.
TEDRAS QUE LIBERAR A LAS TROPAS PRISIONERAS DEL "DICTADOR" EN LAS CUATRO FASES DE ATAQUE, RESCATE, HUIDA Y BATALLA. BEACH HEAD II NO TE DEFRAUDARA.

PIDE ESTOS PROGRAMAS A ERBE Software, SANTA ENGRACIA 17, 28010 MADRID. TFN: (91) 447 34 10 Y EN LAS MEJORES TIENDAS DE INFORMATICA

TENDENCIAS ACTUALES DEL SOFTWARE DE JUEGOS

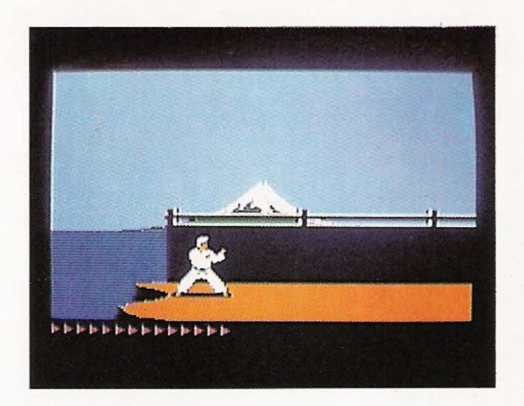
El mundo microinformático gana mayor volumen de usuarios, se mueve más dinero y se crean más intereses. Por ello van sucediéndose distintas ferias a lo largo de la temporada, donde los fabricantes presentan sus nuevos productos. Esto es lo que ocurre en **Consumer Electronic Show (CES)** de Las Vegas, celebrado en enero; el **L.E.T.** de Londres, a mediados de febrero; **Hannover,** a mediados de abril y después el **SICOB,** convocado en París durante el mes de mayo y, nuevamente, el **CES** (pero esta vez en Chicago) y **Hannover,** a mediados de junio. Como clausura, el **SICOB** del otoño, entre los días 18 y 28 de septiembre.

Las más importantes son el **C.E.S.** y **Hannover,** a cada lado del Atlántico, respectivamente.

En tan tumultuosos eventos están presentes las mejores casas de *software* del mundo: **Broderbund, Epyx, Electronics Arts** y un largo etcétera.

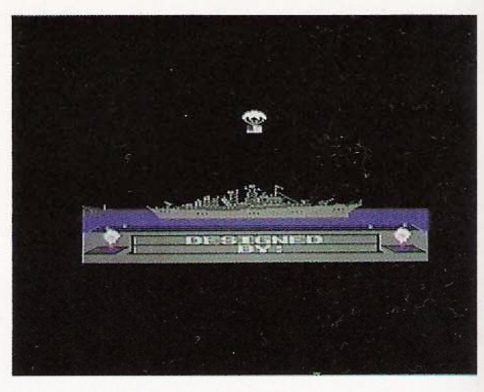
Este año, **Epyx** ha presentado **Summer Games II,** que dispone de las siguientes ocho pruebas: Salto de altura, triple salto, esgrima, canoa, piragua, caballo, bicicleta y jabalina. **Winter Games, The Edilion** y otras colaboraciones llevadas a cabo con **Lucas Film** (de la que también salió **Ballblazer**) son otros productos presentados.

En esta primera feria norteamericana, **Broderbund** presentó su **Karateka,** programa en el cual esta casa continúa haciendo gala de su habitual sofisticación, gráficos cuidados, sonido «sicodélico»..., todo un *best-seller.* No conviene olvidar una serie de programas que fueron marcando esta tendencia de calidad, como el **Print Shop,** que supuso un notable adelanto en programas de manejo de la impresora, aportando nuevas vertientes de creación de nuevos tipos de caligrafía (tecno, rsvp, alexia, etc.) y posibilidades de creación de tarjetas de visita, banderas, cabeceras de carta, que emplean una serie de gráficos que se pueden distribuir por la superficie de las páginas creadas. La experiencia tuvo sus frutos y **Broderbund** sacó un disco de ampliación de gráficos, así como programas que permitían hacer un completo aprovechamiento de los dibujos generados con programas para gráficos (tipo **Doodle I** y **II, Paint Magic, Micro Illustrator,** etc.). Incluso llegaron a lanzarse en el mercado americano paquetes de hojas de colores, especialmente diseñadas para este programa.



Karateka, de Broderbund.

Broderbund lanzó también otra herramienta llamada **Music Shop** que, según el criterio de los entendidos, es el paquete más logrado para composición musical, en el que se combinan las características más relevantes de los programas de mayor aceptación. **Activision** es otra firma cuyos *stands* se abarrotan de público y que en sus comienzos tuvo una tenue aparición en el mercado con su ya desfasado **Decathlon** y que tomó fuerza real con el popular **Gosthbusters.** Desde entonces su tendencia apunta no sólo a los programas de juegos, sino también a las herramientas y utilidades, tales como el **Música Studio,** que tuvo una gran acogida por su planteamiento claramente pedagógico, aunque se tratase de una especie de ensayo. El potencial de esta casa, que trabaja a ambos lados del océano, queda patente con sólo mentar algunos de sus títulos: **Master of the Lamps, Past Finder, Web Dimension, Tour de Francia, Fireworks, Boxing,** etc.



Fire works, de Activision

El papel de algunas casas de *software* europeas también es relevante. **Data Becker** dispone de un buen nivel de conocimientos a nivel *hardware* que le permite desarrollar *software* de verdadera calidad. Un claro ejemplo es un compilador de lenguaje BASIC para el **64,** que compila distintas ampliaciones de este lenguaje de alto nivel. Es una muestra de que lo que no pudo ser en los EE. UU. fue realizado en Alemania. Como dato anecdótico, decir que esta casa obtuvo una importante victoria frente a la «piratería» y de un modo bastante original. Los contratos millonarios pusieron de su lado a uno de los más peligrosos grupos de *crackers* de Europa, autobautizados como **German Cracking Service.** De la noche a la mañana se pusieron a proteger programas. Antes de abandonar sus anteriores ocupaciones hicieron un regalo original: el **An-Zoolook,** que permite proteger programas con una sofisticación tal que hoy en día sigue siendo incopiable un programa protegido por el.

Las firmas europeas continúan ganando puntos, demostrando que pueden dar la talla en los mercados internacionales. Las británicas **Virgin, Gremlin Software,** etc., ponen en circulación programas cada vez más sofisticados.

RUTINAS DE TANTEO Y TIEMPO

- CAMPO DE MINAS
- LA PUNTUACION
- TEMPORIZACION
- EL TECLADO Y EL COMPUTO DEL TIEMPO

No hay nada como saber que sólo dispones de dos segundos para regresar a tu base, o que te faltan diez puntos para obtener la máxima puntuación en tu juego de batallas. Todos los juegos de marcianitos llevan algún tipo de cuenta de tanteo y tiempo, para darle más emoción. Con algunos programas sencillos, tú puedes hacer lo mismo.

Casi todos los juegos de ordenador necesitan algún tipo de puntuación o temporización, o incluso ambas cosas. Sin ellos no puedes juzgar lo bien que se te da el juego, o si vas mejorando algo, y no suele tener mucho interés jugarlo con tus amigos.

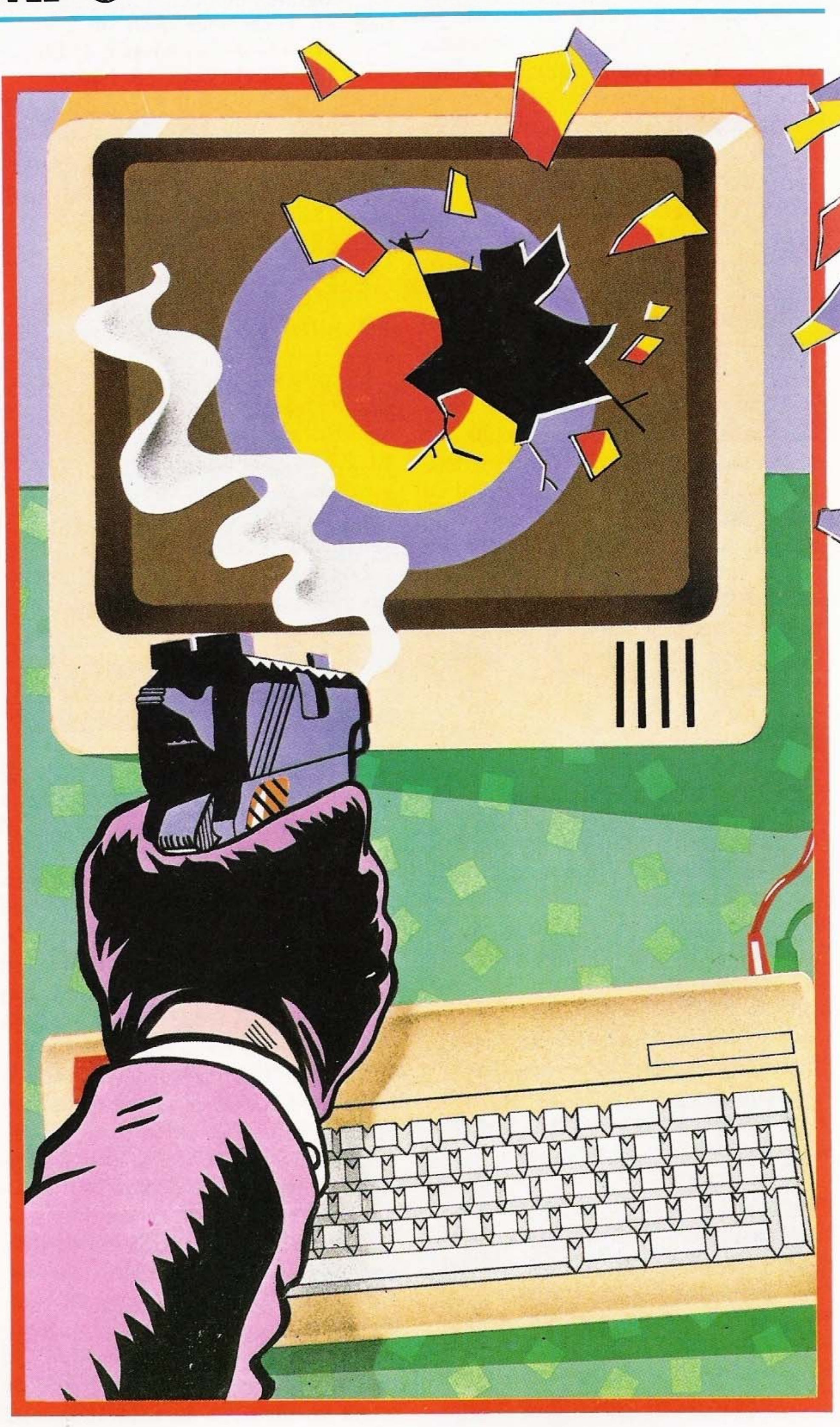
Podrías tener a alguien sentado tras de ti que fuera contando los impactos que consigues sobre tu feroz enemigo, pero esto no tiene mucho sentido cuando puedes programar a tu ordenador para que los cuente él. Con unas cuantas líneas de programa más, la máquina recordará también las puntuaciones.

Por la misma razón no hay necesidad de recurrir a un cronógrafo para la medida del tiempo. Todas las máquinas llevan un reloj incorporado, y puedes servirte de él de muchas formas para mejorar tus juegos.

CAMPO DE MINAS

Para que veas la manera de incorporar en la práctica las rutinas de puntuación y temporización, aquí tienes un juego en el que se van añadiendo las rutinas progresivamente. Cada rutina es muy sencilla y se puede añadir también a otros juegos.

El juego se llama Campo de Minas, y en él tú vas conduciendo un carro de combate, cuya misión es rescatar a unos paracaidistas que se han arrojado temerariamente sobre un campo minado. Cada vez que el tanque se



mueve, corre el riesgo de hacer detonar una mina plantada aleatoriamente por el ordenador. Como en un campo minado de verdad, las minas son invisibles, por lo que tendrás que moverte con precaución.

El tanque (desafortunadamente es sólo un signo #, hasta que aprendas la manera de combinar gráficos y movimiento en un programa BASIC) se controla utilizando las teclas normales de movimiento: Z para izquierda, X para derecha, P hacia arriba y L abajo. De hecho, el núcleo del programa está constituido por la rutina de "moverse por la pantalla", que ya conoces.

Cuando teclees esta sección del juego y la ejecutes (con RUN), verás que todavía no está completa: después de que hayas rescatado al paracaidista, no sucede nada, excepto que el tanque continúa vagando sin rumbo por la pantalla. El programa ha de detenerse pulsando la tecla RUN/STOP o tendrás que esperar hasta que el tanque tropiece con una mina escondida. Pero no te alarmes, todo esto mejorará en cuanto le pongas las rutinas de puntuación y de tiempo que siguen.

Teclea para Commodore-64

```
 15 Z$="■■■■■■■■■
    ■■■■":POKE 65Ø,128
 5Ø LET PO=2Ø5
 6Ø PRINT "◘";Z$;"---------
    ----------------------
    ---------"
 7Ø POKE 5328Ø,1:POKE
    53281,3
 85 LET L=5
 9Ø LET X=INT(RND(1)*479)+1
11Ø IF X=PO THEN GOTO 9Ø
115 POKE 1Ø24+X,87
12Ø POKE 1Ø24+PO,35
13Ø LET LP=PO: LET LL=L:
    GET A$
14Ø IF A$="Z" THEN LET
    PO=PO-1: L=L-1:GOTO 19Ø
15Ø IF A$="X" THEN LET
    PO=PO+1: L=L+1:GOTO 21Ø
16Ø IF A$="P" THEN LET
    PO=PO-4Ø: GOTO 22Ø
17Ø IF A$="L" THEN LET
    PO=PO+4Ø: GOTO 22Ø
18Ø GOTO 13Ø
19Ø IF L<Ø THEN LET PO=LP:
    LET L=LL
2ØØ GOTO 22Ø
21Ø IF L>39 THEN LET PO=LP:
    LET L=LL
22Ø IF 1Ø24+PO<1Ø24 OR
    1Ø24+PO>1Ø24+479 THEN
    LET PO=LP: GOTO 13Ø
23Ø POKE 1Ø24+LP,32
24Ø POKE 1Ø24+PO,35
25Ø LET M=INT(RND(1)*479)+1
27Ø IF M=PO THEN POKE
    1Ø24+PO,432:PRINT
    Z$;"■BOOM!!-HAS
    ENCONTRADO UNA MINA":
    STOP
31Ø GOTO 13Ø
```

En este programa, la pantalla queda dividida en dos mitades por una raya trazada por la línea de programa 60. Z$ ha sido definida en la línea 15 y es una cadena de controles de cursor. El POKE de la línea 15 pone en marcha la repetición de tecla.

Los colores de la pantalla y márgenes se establecen en los POKEs de la línea 70.

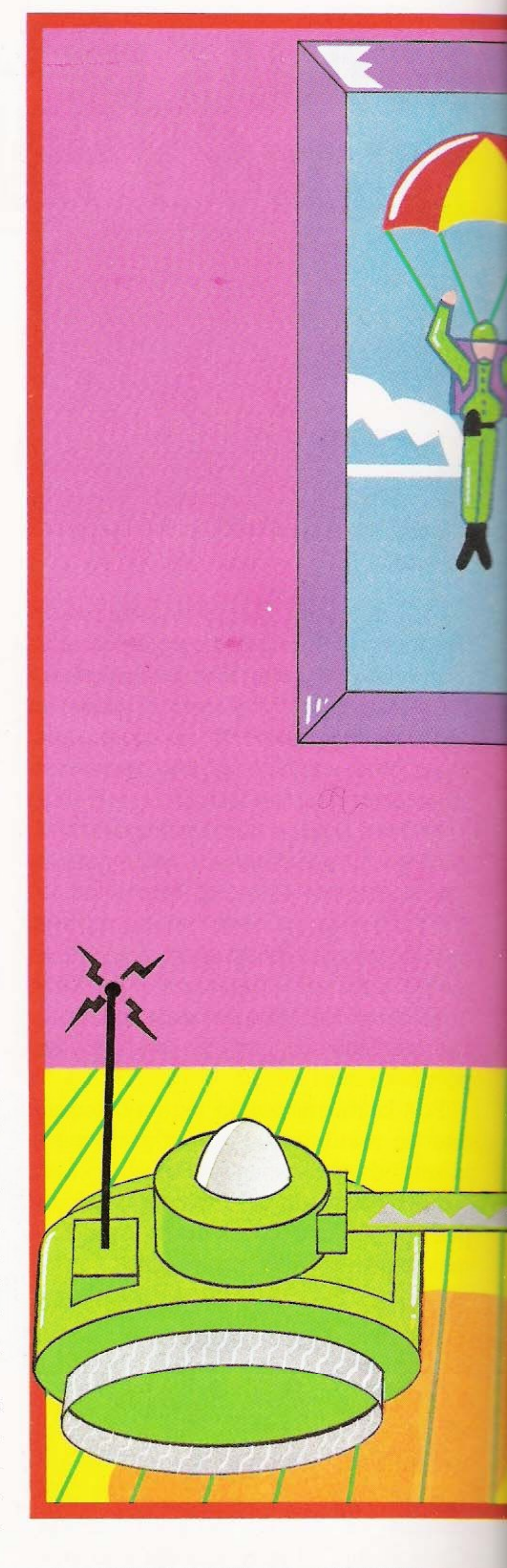
La línea 85 define L, que se usa después en las líneas 190 y 210 para evitar que el tanque se salga por algún lado de la pantalla.

Para representar los caracteres en la pantalla se utilizan POKEs. PO es la posición del tanque, y la línea 120 lo dibuja en la pantalla.

La posición donde cae el paracaidista se selecciona en la línea 90. Esta posición se compara con la posición del tanque. Si ambas coinciden, se elige una nueva posición de caída para el paracaidista. La línea 115 dibuja el paracaidista en la pantalla.

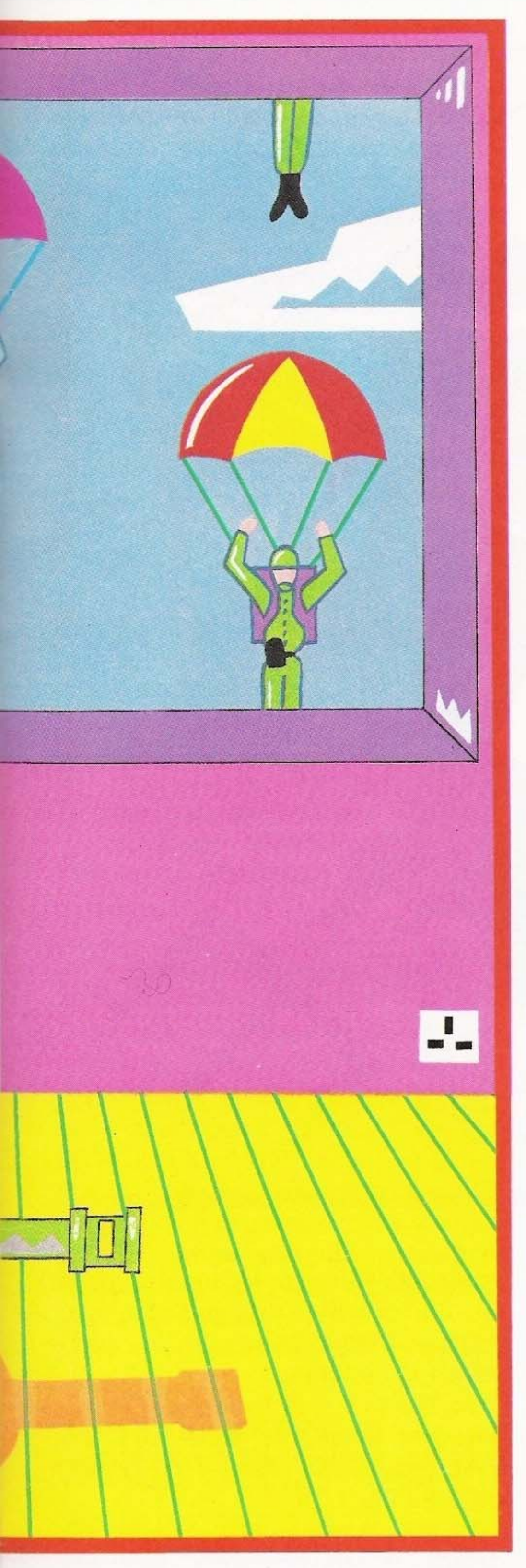
La línea 220 impide que el tanque salga por la parte de arriba de la pantalla o por debajo de la línea de trazos. La línea 230 borra la última posición del tanque y la línea 240 lo representa en su nueva posición.

Finalmente, la línea 250 selecciona una posición aleatoria para la mina. En la línea 270 se comparan las posiciones de la mina y del tanque. Cuando ambas posiciones coinciden, el tanque es borrado con un POKE, y se envía un mensaje de explosión.



PUNTUACION

En los juegos de extraterrestres, la puntuación aumenta normalmente cuando la posición ocupada por dos objetos en la pantalla es la misma.



Los objetos pueden ser un misil y un blanco, un comecocos y una píldora alimenticia, un tanque y un paracaidista, o lo que el juego requiera.

Añade, pues, estas líneas a tu programa para ver cómo trabaja en la práctica el mecanismo de tanteo.

```
 40 LET S=0
280 IF X=PO THEN LET S=S+1:
    GOTO 90
330 POKE53280,254:POKE53281,
    246: PRINT
    "■□□□□□□□□";S;
    "PARACAIDISTAS"
```

Cambia el STOP de la línea 270 por un GOTO 330. La línea 270 se convierte ahora en:

```
270 IF M=PO THEN POKE
    1024+PO,32:PRINT
    Z$;"■BOOM!!-HAS CHOCADO
    CON UNA MINA":GOTO 330
```

La línea 280 es importante. En ella se comprueba que el tanque y el paracaidista ocupan la misma posición en la pantalla. Si ocurre esto, la puntuación aumenta en 1.

En la línea 40 se pone a cero el tanteo antes de que empiece el juego y la línea 330 visualiza la puntuación. Cambiando el STOP de la línea 270 se consigue que el ordenador presente la puntuación después de haber tocado una mina.

El jugador se enfrenta ahora con una sucesión de paracaidistas a los que rescatar. Cada vez que es rescatado uno, cae otro del cielo. El juego se detiene cuando se produce la explosión de una mina (porque el tanque ocupa la misma posición en la pantalla).

PUNTUACION MAXIMA

No es difícil añadirle a tu juego una opción de puntuación máxima. No tienes más que introducir una variable asociada a esta puntuación —por ejemplo, HS— y algún método para actualizarla cuando sea superada dicha opción, además hay que poner una rutina de presentación.

Aquí tienes las líneas que debes añadir para tener una opción de puntuación máxima.

```
 30 LET HS=0
350 IF S>HS THEN LET HS=S
370 PRINT "■□□□□□□□□□
    MAYOR PUNTUACION";HS
```

En primer lugar debes poner la puntuación máxima en su menor valor posible, por lo que la línea 30 pone HS a cero. Después de que el juego se ha detenido, la línea 350 compara la última puntuación (S) con la puntuación máxima (HS). Si la anterior puntuación obtenida es mayor que la puntuación máxima se actualiza HS, haciéndola igual a S. Finalmente, la línea 370 visualiza en la pantalla el valor.

Es probable que estas líneas te parezcan suficientes para dotar el juego de un tanteo. Por desgracia, esto no es cierto. Cada vez que le hagas ejecutar el programa (con RUN), el ordenador olvida automáticamente el valor de HS, y los valores de las otras variables. Para mantener el valor de HS tienes que añadir las líneas de "¿Otra vez?" que se describieron en un capítulo anterior.

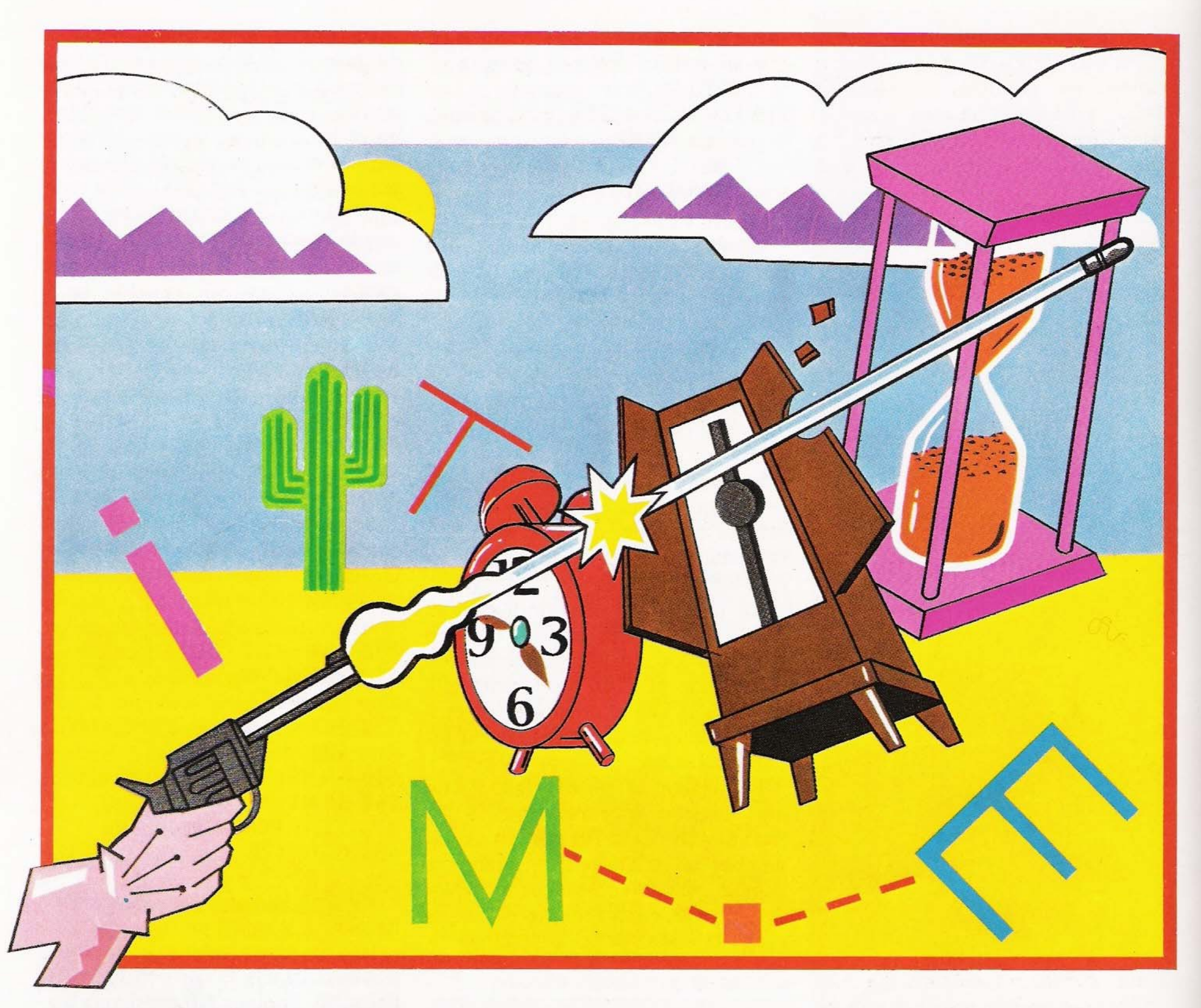
```
390 FOR F=1 TO 1000:NEXT F
410 PRINT "■□□□□□□□□□
    OTRA VEZ? (S/N)";
420 GET K$:IF K$=""THEN GOTO
    420
430 IF K$="S"THEN GOTO 40
440 IF K$="N" THEN PRINT
    "□": POKE 650,0:END
450 GOTO 420
```

He aquí lo que hacen estas nuevas líneas:

Hay un corto retardo introducido por la sentencia FOR... NEXT en la línea 390. La línea 400 borra una parte de la pantalla dejándola preparada para el mensaje «Otra vez» (S/N) que se presenta en la línea 410.

La rutina "Otra vez" está en las líneas 410 a 450. La línea 430 hace que el programa recomience en la línea 40 si se pulsa S, y la línea 440 detiene el programa si se pulsa N. La línea 450 sirve para asegurarse de que cualquier otra tecla será ignorada.

Como no hay necesidad de pulsar RUN cada vez que desees jugar de nuevo, el valor de HS será preservado, aunque al volver a cargar el programa (con LOAD) se perderá el valor de HS, incluso si encuentras alguna forma de arrancar el programa sin pulsar RUN.



MEDIDA DE TIEMPOS

Tal como está el juego, depende demasiado de la suerte, simplemente, el jugador sigue adelante hasta que pisa una mina escondida.

Se puede introducir en este tipo de juegos un elemento de habilidad, para convertirlo en una carrera contra el reloj. Con los siguientes añadidos puedes cronometrar cuánto tardas en rescatar a diez paracaidistas.

```
 80 LET TI$="000000":L=5
290 IF S<10 AND X=PO THEN
    GOTO 90
300 IF S=10 THEN GOTO 320
320 LET
    T=VAL(RIGHT$(TI$,2))+60*
    VAL(MID$(TI$,3,2))
340 IF S=10 THEN PRINT
    "■□□□□□□□□□IN";T;
    "□SEGUNDOS"
```

El reloj interno de la máquina está corriendo todo el tiempo que el ordenador permanece encendido. Para arrancar el contador de tiempos, tienes que poner a cero la lectura de dicho reloj. Esto se hace en la línea 80. Simplemente teclea LET TI$= "0000".

El reloj se «detiene» en la línea 320. Realmente, el reloj no puede detenerse; lo que tú haces es que la máquina recuerde una lectura particular en un instante determinado, por ejemplo cuando coinciden dos objetos en la pantalla.

Se llama T a la lectura del contador, por lo que se pone LET T= VAL [RIGHT$ (TI$,2) + 60 * VAL (MID$ (TI$,3,2)].

El reloj funciona actualizando una cadena de seis caracteres, tal como establece la línea 80. Empezando por la derecha, la cadena contiene dos dígitos para las horas, dos para los minutos y dos para los segundos. En la línea 320 se obtiene un valor de la cadena para tener una lectura en segundos. El número de minutos que han transcurrido se multiplica por 60

y se suma al número de segundos. MID$ (TI$,3,2) sirve para mirar el número de minutos, y RIGHT$ (TI$,2), para ver el número de segundos. (Una vez más, en un próximo artículo examinaremos a fondo este tipo de sentencias.)

El reloj debe detenerse cuando el jugador ha rescatado a diez paracaidistas, por lo que en la línea 300 se comprueba si ya se han rescatado diez, en cuyo caso, el programa salta a la línea 320, que es la que "detiene" al reloj. La línea 340 imprime el tiempo que se ha tardado en rescatar a diez paracaidistas.

Si se ha rescatado con éxito a un paracaidista **y, además,** el número total de los que van ya rescatados es menor que diez, la línea 290 hace caer otro paracaidista.

La línea 340 presenta el tiempo transcurrido para el rescate sólo si se han rescatado ya los diez. La lectura de tiempo es dividida por 50, por lo que el tiempo aparece en segundos. El reloj es actualizado 50 veces cada segundo.

MEJOR TIEMPO

De la misma forma que añadiste antes una opción de tanteo máximo para el juego, puede resultarte interesante una opción de mejor tiempo. En esta variante del juego tienes que poder registrar el tiempo más rápido en que los diez paracaidistas son rescatados, aunque este principio se puede aplicar a cualquier temporización que desees hacer.

He aquí las líneas que tienes que añadir:

```
20 LET LT=999999
360 IF T<LT AND S=10 THEN LET
    LT=T
380 PRINT"■□□□□□□□□□
    LOW TIME=";LT;
    "□SEGUNDOS"
```

Igual que con el tanteo máximo se ponía inicialmente un "tanteo máximo" muy bajo, ahora se pone un "tiempo récord" ridículamente largo. La línea 20 asigna a la variable de mejor tiempo (LT) un valor de 999999.

La línea 360 compara el último tiempo obtenido con el mejor tiempo. Si el último tiempo obtenido es más corto que el tiempo récord **y, además,** se han rescatado ya diez paracaidistas, entonces se modifica el tiempo récord haciéndolo igual al último tiempo obtenido.

Finalmente, la línea 380 sirve para expresar en segundos el tiempo récord. La variable de tiempo récord es dividida por 50 para que resulte en segundos.

Una cosa que has de recordar es que si estás utilizando una opción de tiempo récord en un juego, tienes que utilizar la rutina de "¿Otra vez?", pues, de lo contrario, el valor del tiempo récord se perderá cada vez que ejecutes (con RUN) el programa.

EL TECLADO Y LA CUENTA DEL TIEMPO

Hasta ahora has visto cómo puedes controlar el reloj interno de la máquina desde dentro de un programa, examinando las posiciones de dos objetos sobre la pantalla. Otra forma de "detener" el reloj es servirte del teclado.

Puedes hacerlo con la sentencia GET$. Resulta tan fácil como arrancar y parar un cronómetro para controlar el movimiento de los objetos por la pantalla.

Aquí tienes un juego de acción rápida que ilustra cómo puede usarse el teclado para detener el reloj:

```
20 PRINT"□"
30 LET N=INT(RND(1)*900)+1
40 FOR F=0 TO N
50 NEXT F
60 PRINT "□■■■■■■■";
   TAB(16); "TIRA!!"
70 LET TI$="000000":POKE
   198,0
80 GET K$:IF K$=""THEN GOTO
   80
90 LET T=TI
100 PRINT "■■■■■■■■";
    TAB(16); "BANG!!"
110 FOR F=1 TO 300
120 NEXT F
130 LET M=INT(RND(1)*35)+1
140 IF T<M THEN PRINT
    "■■■■■■■■■■"TAB(51)
    "HAS SOBREVIVIDO"
150 IF T>M THEN PRINT
    "■■■■■■■■■■"TAB(53)
    "HAS MUERTO"
160 IF T=M THEN PRINT
    "■■■■■■■■■■"TAB(50)
    "HABEIS MUERTO"
```

El programa presenta el mensaje "DISPARA!!" y el jugador ha de pulsar cualquier tecla tan rápido como pudea. Se mide el tiempo de reacción desde el momento en que apareció el mensaje.

Las líneas 30 a 50 introducen una pausa aleatoria. La línea 60 sirve para enviar el mensaje "DISPARA!!" e inmediatamente se arranca el contador de tiempo en la línea 70. La línea 80 hace que la máquina espere, continuando cuando se ha pulsado una tecla cualquiera. Ya vimos esta línea al ocuparnos del "Control del Teclado".

En cuanto se ha pulsado una tecla cualquiera, la línea 90 para el contador, llamado T a la lectura que tiene en ese momento. La línea 100 "BANG!!". Hay una pausa introducida por las líneas 110 y 120 antes de que la máquina elifa un instante de tiro. La línea 130 es la que hace esto.

La máquina tiene ahora dos variables, tu tiempo, T, y el tiempo de la máquina, M. Las líneas 140 a 160 comparan estos valores y presentan el resultado del duelo.

P y R

¿Existe algún límite para la duración máxima que se puede tener?

Si, existe un límite, aunque normalmente es tan alto que en la práctica no tiene importancia. El reloj interno de casi todos los ordenadores domésticos avanza a la misma velocidad, y el factor limitador es la cantidad de pulsos de tiempo que el ordenador puede recordar. En **Comodore** se pueden tener hasta dos bytes (65535) lo que equivale a unos 22 minutos.

JUEGOS DE LABERINTO

- CREACION DE LABERINTOS
- PUNTUACION Y TIEMPOS
- COMO SE DESPLAZA UN CARACTER
- DISEÑOS PROPIOS

Los juegos de laberintos sofisticados requieren programas largos. Pero tú puedes diseñar algunos sencillos extrayendo de ellos importantes principios y utilizando poco más que un bucle y sentencias DATA.

Los juegos de laberintos ejercen una fascinación permanente sobre los propietarios de un ordenador, por lo que las casas de *software* continúan sacando nuevas variantes del comecocos.

Este artículo te enseñará la manera de saltar al carro de los fabricantes de laberintos, permitiendo que te construyas el tuyo propio.

En su primera fase el laberinto no incluye «enemigos» ni obstáculos, ya que esto requeriría un programa muy largo. Pero te enseñará cómo se programa el que tu carácter principal no pueda atravesar las paredes, lo cual es la base de todos los juegos de esta clase. También se incluye la puntuación y el crono, así como una rutina de "mejor tanteo", a fin de darle un cierto interés competitivo.

Este laberinto para el **C-64** se construye a base de sentencias PRINT y de los gráficos en ROM, en una rutina que se extiende desde la línea 60 a la 170 en el programa que va más adelante.

Al ejecutar este programa, aparece un laberinto predefinido hacia la mitad de la pantalla. Utiliza las teclas Z y X para mover el carácter π hacia la izquierda y la derecha respectivamente, y las teclas P y L para moverlo arriba y abajo. Los puntos —en este juego "monedas"— se van borrando a medida que el π va pasando sobre ellos, llevando al mismo tiempo la cuenta.

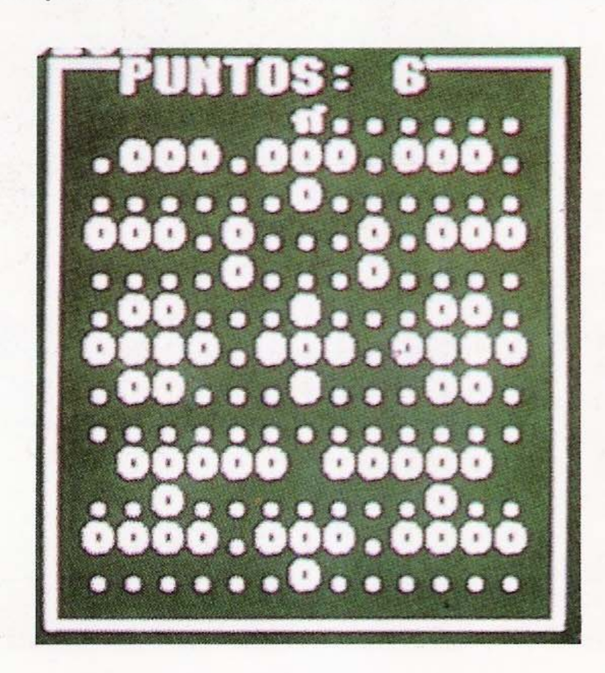
De esta forma el programa puede que no resulte muy emocionante, pero muestra claramente la manera de construir juegos de laberintos.

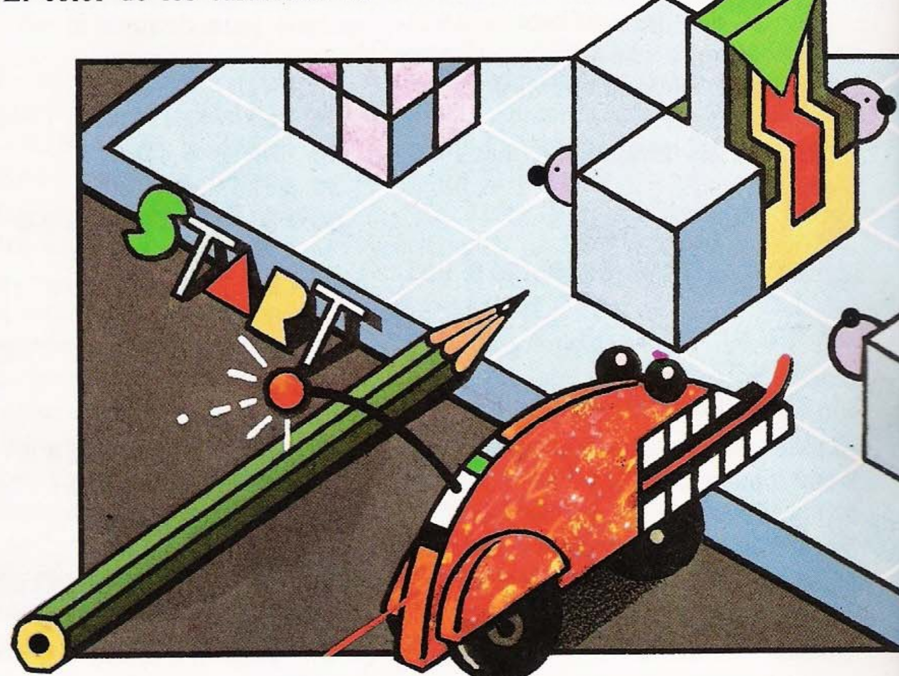
Mirando al programa, puedes ver que la primera parte, línea 10 a 40, se utiliza para establecer las variables. P1 es la posición de memoria correspondiente al punto de la pantalla en el que empiezan las "monedas" del laberinto. BD es la dirección del contorno y BG es el color de fondo de la pantalla, que en las líneas 45 y 50 cambia a azul o negro, respectivamente. El color de los caracteres se define en las líneas 30: alterando el valor que hay entre paréntesis por un código de color adecuado (consulta el apándice de tu manual para ver los códigos ASCII y CHR$) puedes modificar a tu gusto el color del laberinto.

La línea 35 muestra otra manera de programas el efecto de CLR/HOME. Anteriormente se utilizó el símbolo del corazón inverso como parte de una sentencia PRINT, para borrar la pantalla y situar el cursor en la parte superior izquierda, listo para recibir la instrucción siguiente. Puedes utilizar CHR$ (147) exactamente de la misma forma, pero en un programa largo donde es importante ahorrar memoria, es más conveniente definir como una variable de cadena, tal como se hace en la línea 35.

La línea 55 hace posible la autorrepetición de la tecla, que resulta esencial en un juego laberíntico de este tipo.

A continuación viene la secuencia de dibujo del laberinto, a base de





sentencias PRINT, seguida en la línea 200 por un POKE para situar el signo π en la dirección 1117 de memoria, la esquina superior izquierda del laberinto. Seguidamente el programa envía un mensaje en el que pide una pulsación de tecla para que el juego empiece (o continúe). La línea 225 contiene dos controles de cursor hacia arriba y cuarenta espacios en blanco dentro de una sentencia PRINT. En cuanto se pulsa [SPACE], se imprime la línea 225 encima del mensaje de pulsación de tecla, es ésta una rutina muy sencilla de borrado y puede utilizarse en muchos tipos de programa. Otra manera de programar esto sería:

```
225 PRINT "□□□":FOR T=Ø TO
    39: PRINT "□";:NEXT T
```

Como puedes ver, con un sencillo bucle FOR...NEXT se puede hacer lo mismo de un modo más elegante.

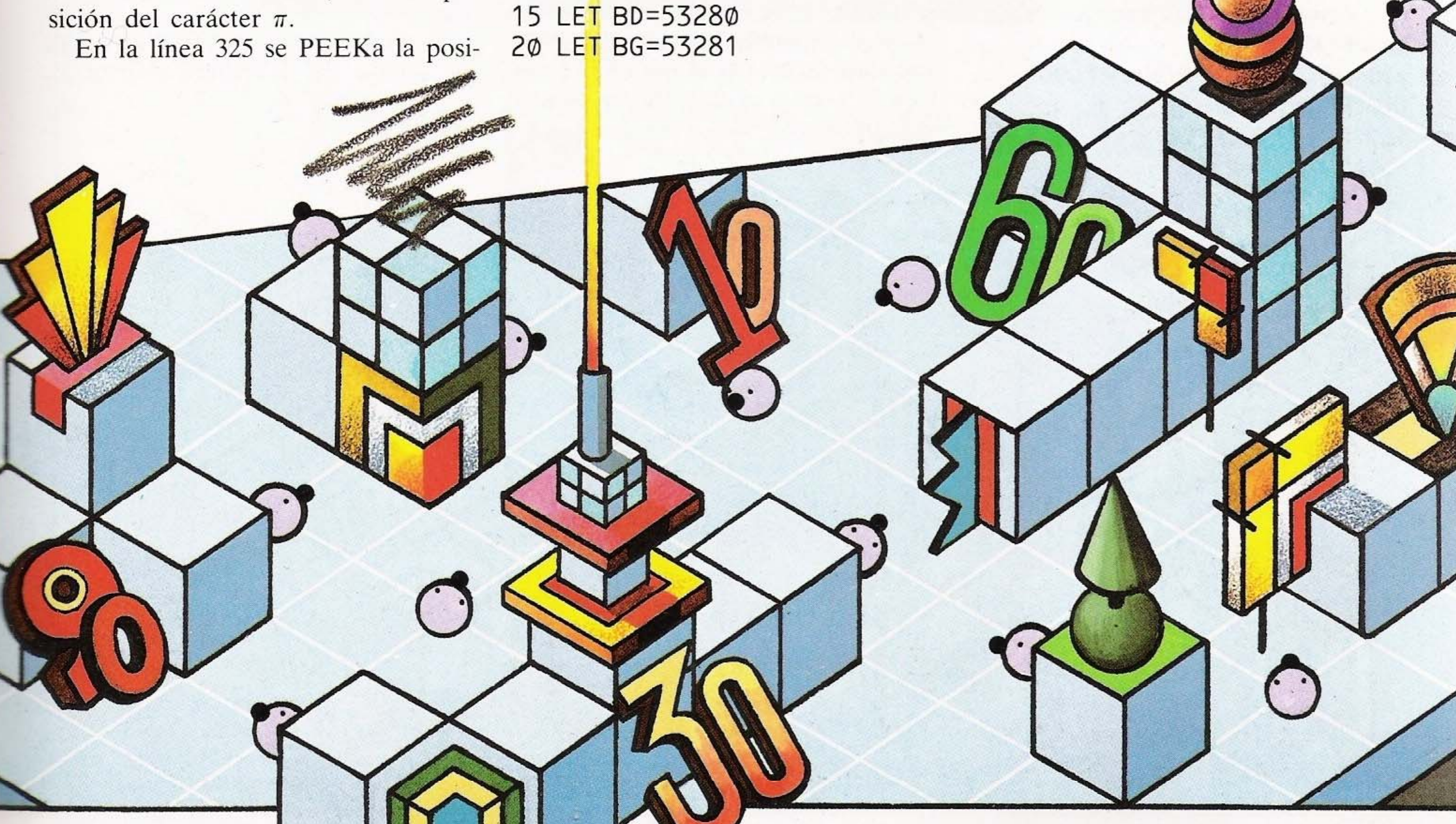
En las líneas 300 a 320 está la rutina detectora de pulsaciones en el teclado, que utiliza un GET y una sucesión de sentencias IF...THEN, para establecer el valor de P2, la nueva posición del carácter π.

En la línea 325 se PEEKa la posición de memoria que contiene la dirección de π, y hace un GOTO a la visualización de la conteo de monedas, si dicha posición no contiene un espacio ni una moneda. Si contiene moneda, se incrementa en 1 el registro de cuenta de monedas (línea 330), se borra la posición anterior de π, P1 (línea 335), y se establece su nuevo valor (línea 340). En efecto, la línea 325 evita que se escriba encima de los caracteres del laberinto.

El programa sigue visualizando la cuenta de monedas y la rutina GET se repite hasta que CO es igual a 103, en la línea 365, en cuyo caso el programa vuelve a ejecutarse automáticamente.

Para adaptar el programa a el VIC-20 hay que realizar algunos cambios. Omite las líneas 15 y 45. En la línea 10, utiliza 7735 en lugar de 1117. En la línea 20, utiliza 36879 en lugar de 53281. En la línea 25, utiliza 22 en lugar de 40. En la línea 50, emplea 30 en lugar de 0. En la 225 inserta tres espacios en blanco. En la línea 260, cambia TAB(14) por TAB(4).

```
10 LET P1=1117: LET P2=P1
15 LET BD=53280
20 LET BG=53281
25 LET LL=40
30 LET CC$=CHR$(5)
35 LET CS$=CHR$(147)
40 LET CH$=CHR$(19)
45 POKE BD,6
50 POKE BG,0
55 POKE 650,128
60 PRINT CS$,CC$
100 PRINT TAB(12)
    "□□□□□□□□□□□□□□□"
105 PRINT TAB(12)
    "□.............□"
110 PRINT TAB(12)
    "□.OOO.OOO.OOO.□"
115 PRINT TAB(12)
    "□......O......□"
120 PRINT TAB(12)
    "□OOO.O...O.OOO□"
125 PRINT TAB(12)
    "□....O...O....□"
130 PRINT TAB(12)
    "□.OO...●...OO.□"
135 PRINT TAB(12)
    "□O●●O.●O●.O●●O□"
140 PRINT TAB(12)
    "□.OO...●...OO.□"
145 PRINT TAB(12)
    "□.............□"
```



```
150 PRINT TAB(12)
    "□OOOOO.OOOOO□"
155 PRINT TAB(12)
    "□..O.......O..□"
160 PRINT TAB(12)
    "□OOOO.OOO.OOOO□"
165 PRINT TAB(12)
    "□......O......□"
170 PRINT TAB(12)
    "□□□□□□□□□□□□□□□□□"
200 POKE P1,94
210 PRINT "■FIN DEL JUEGO.
    PULSA LA BARRA
    ESPACIADORA PARA
    EMPEZAR."
215 GET K$
220 IF K$<>"□" THEN 215
225 PRINT "OO□□□□□□□□
    □□□□□□□□□□□□
    □□□□□□□□□□□
    □□□□□□□□□□□□"
300 GET A$
305 IF A$="Z" THEN P2=P1-1
310 IF A$="X" THEN P2=P1+1
315 IF A$="P" THEN P2=P1-LL
320 IF A$="L" THEN P2=P1+LL
325 IF PEEK(P2)<>32 AND
    PEEK(P2)<>46 THEN GOTO
    350
330 IF PEEK(P2)=46 THEN
    CO=CO+1
335 POKE P1,32
340 LET P1=P2: POKE P1,94
350 PRINT CH$;
360 PRINT TAB(14);
    "PUNTO:";CO
365 IF CO<103 THEN 300
400 POKE BG,6: CO=0
415 FOR T=1 TO 2000: NEXT T:
    GOTO 10
415 FOR T=1 TO 2000: NEXT T:
    GOTO 10
```

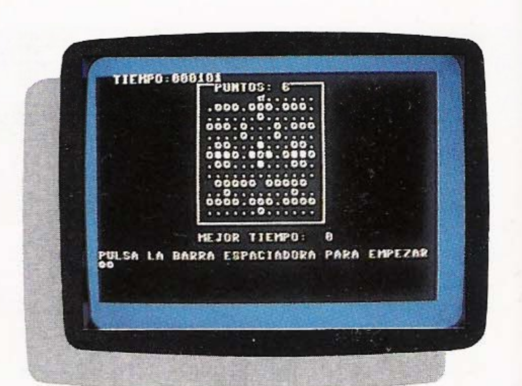
UN POCO MAS COMPETITIVO

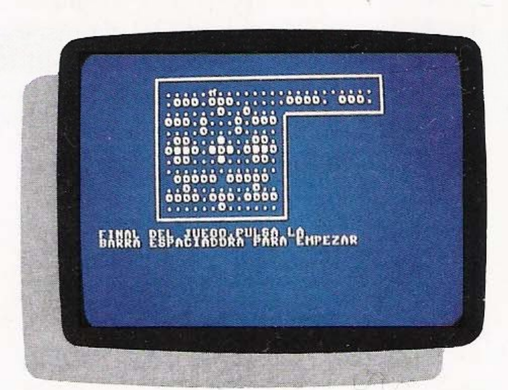
Añádele ahora las líneas de cronometraje:

```
205 PRINT TAB(12);"■MEJOR
    TIEMPO: ";BT
230 LET TI$"000000"
355 PRINT TAB(2);"TIEMPO:
    ";TI$;
400 IF G=0 THEN BT=VAL(TI$):
    G=1
405 IF VAL(TI$)<BT THEN
    BT=VAL(TI$)
410 LET CO=0: POKE BG,6
```

CAMBIO DEL LABERINTO

Si quieres cambiar el laberinto, ajusta el contenido de las sentencias PRINT en las líneas 105 a 165. Puedes usar cualquiera de los símbolos incorporados en ROM, e incluso modificar el tamaño del laberinto, pero acuérdate de ajustar el valor CO en la línea 365, para el mínimo de puntos de tu laberinto.





Arriba, foto de la pantalla, tal como resulta del programa explicado en el texto. Sin embargo, con un poco de imaginación, se puede cambiar el diseño del laberinto en el listado y darle así cualquier aspecto (abajo).

EL LENGUAJE DE LA TORTUGA (II)

- EDICION
- DISEÑO DE PROCEDIMIENTOS
- COMPROBACION DE LA MEMORIA
- DIBUJANDO UN HEXAGONO

Estrellas, círculos, espirales y hexágonos, mágicamente dibujados en tamaños continuamente crecientes. Son sólo algunas de las formas que puedes conseguir con unas cuantas instrucciones utilizando el LOGO.

En la primera parte de este artículo vimos cómo dibujar con la tortuga del LOGO, cómo crear una primitiva de LOGO enseñando a la tortuga a hacer un dibujo, y cómo utilizar estas primitivas para ayudar a definir nuevas primitivas. Por ejemplo, es posible enseñar a la tortuga a dibujar un hexágono:

```
TO HEXAGONO
REPEAT 6 [FORWARD 7Ø RIGHT
6Ø]
END
```

Después de teclear END, el LOGO te presentará el mensaje HEXAGONO DEFINED para que sepas que, a partir de ahora, HEXAGONO forma parte de su vocabulario. Cada vez que teclees HEXAGONO, la tortuga dibujará un hexágono y terminará apuntando en la misma dirección que tenía cuando empezó:

Puedes utilizar HEXAGONO para definir MODELO:

```
TO MODELO
REPEAT 12 [HEXAGONO FORWARD
1Ø RIGHT 3Ø]
END
```

EL CAMPO VISUAL DE LA TORTUGA

La geometría de la tortuga es diferente a la geometría de coordenadas, que, probablemente, te resulta más familiar, donde las posiciones se definen con referencia a un punto externo. (Las coordenadas normales de la pantalla en los gráficos de tu ordenador se definen de esta forma.) La geometría de la tortuga funciona de manera diferente, ya que el sistema de referencia está en la propia tortuga. Al dibujar un hexágono, la tortuga gira 60 grados y no 120, aunque las dos líneas que la tortuga ha dibujado formen entre sí un ángulo de 120 grados. Si tienes algún problema para entender el sistema de referencia de la tortuga, imagínate que vas andando por la figura que quieres dibujar. Los ángulos y distancias que tienes que describir y recorrer son los que has de traducir en instrucciones para la tortuga.

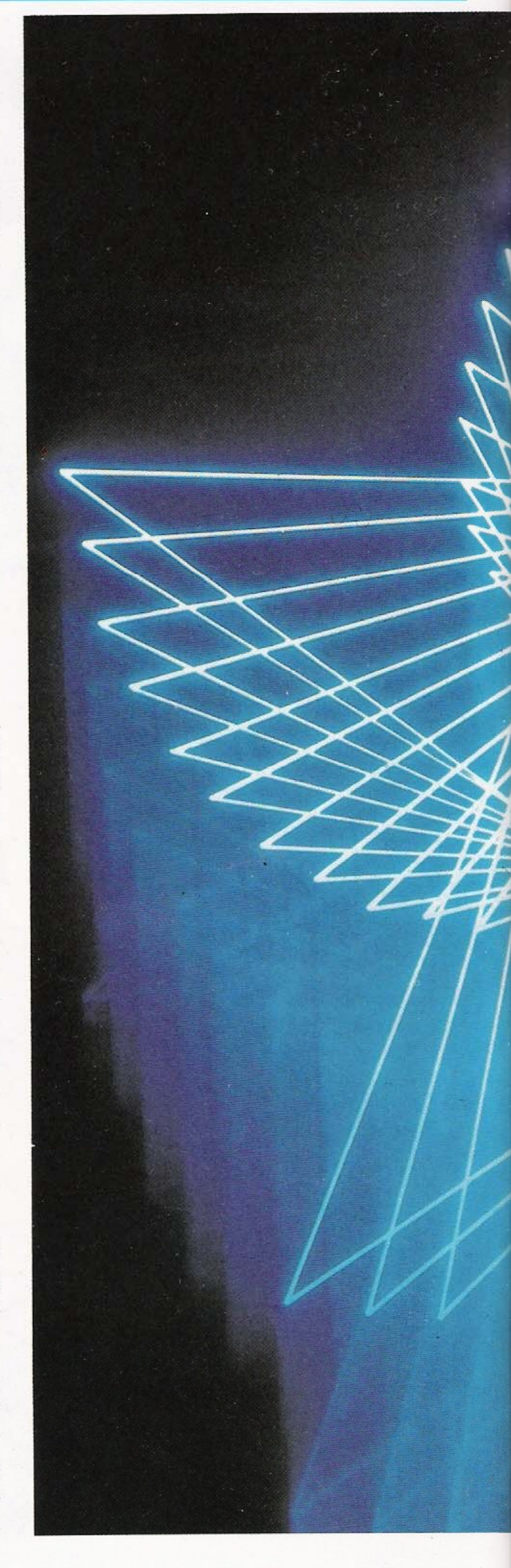
Aunque esto puede parecer confuso, cuando te hayas acostumbrado te resultará realmente fácil, como suele suceder a los niños, ya que se relaciona directamente con su propia experiencia.

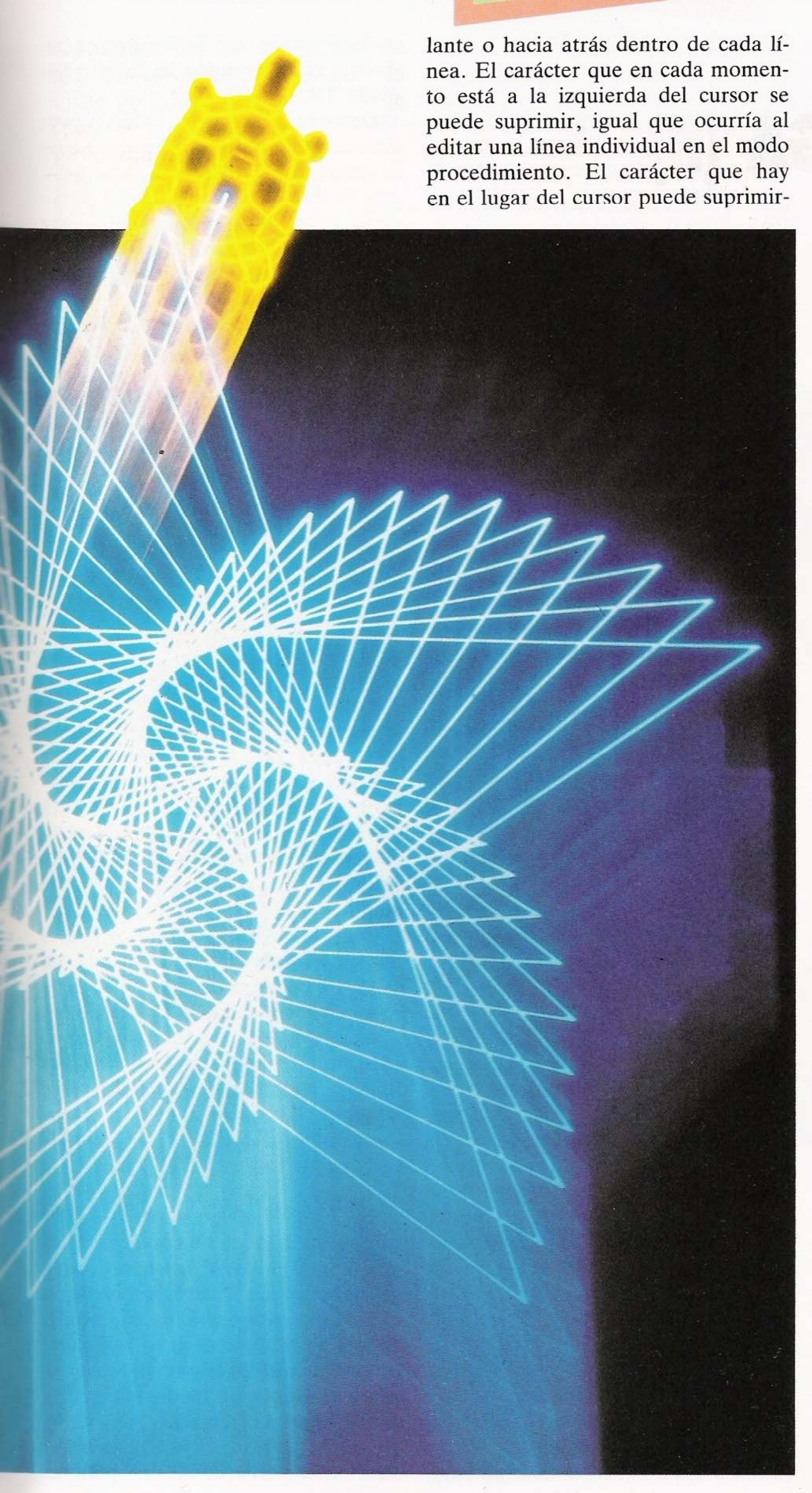
EDICION

Cuando los errores de programación empiezan a hacer estragos, hay que eliminarlos del programa. Para poder hacer esto, hay que ir a un tercer modo, el *Editor*. El modo de edición es semejante al modo de procedimiento. No afecta al estado inmediato de la tortuga.

Para editar un procedimiento, teclea EDIT" y el nombre del procedimiento. En algunas versiones del LOGO no hacen falta las comillas. En lugar de EDIT se puede utilizar la abreviatura ED.

Si tecleas EDIT "HEXAGONO habrás entrado en el modo EDIT. En la parte superior de la pantalla aparecerá la definición de HEXAGONO. En el modo EDIT no hay gráficos, toda la pantalla está dedicada al texto. Utilizando las teclas de desplazamiento del cursor, puedes mover éste hacia arriba o hacia abajo de una línea de programa a otra, o hacia de-





lante o hacia atrás dentro de cada línea. El carácter que en cada momento está a la izquierda del cursor se puede suprimir, igual que ocurría al editar una línea individual en el modo procedimiento. El carácter que hay en el lugar del cursor puede suprimirse con la tecla de borrado *Delete* y la de Control. Para insertar caracteres no hay más que teclearlos, el resto de la línea ya se irá moviendo para hacerles sitio.

Aunque los procedimientos sean los mismos, las teclas pueden diferir según cada tipo de ordenador. En **Commodore,** CTRL A sitúa el cursor al principio de una línea y CTRL L lo lleva al final de la línea. CTRL K (del inglés *Kill* = matar) borra todos los caracteres que hay a la derecha del cursor. Para borrar una línea entera, utiliza CTRL A y a continuación CTRL K. Con CTRL O se abre un espacio para que se pueda introducir una nueva línea.

En el modo EDIT se puede definir un nuevo procedimiento. La ventaja de hacer esto es que puedes cambiarte de una línea a otra para modificar instrucciones mientras todavía estás definiendo el procedimiento. Para definir Flor de esta manera tienes que teclear:

```
EDIT "FLOR
```

En la parte superior de una pantalla vacía aparecerá TO FLOR. Puedes definir ahora el procedimiento de la forma corriente, aprovechando las ventajas que te brinda el editor de pantalla.

Cuando se define un procedimiento, el modo de edición te será accesible tanto si teclea TO como si tecleas EDIT.

DALE LAS ENTRADAS CORRECTAS

Existe una importante diferencia entre HEXAGONO y RIGHT. Con HEXAGONO siempre dibujará un hexágono de 60 unidades de lado. En cambio, RIGHT requiere un valor de entrada que diga a la tortuga cuánto tiene que girar a la derecha; de la misma forma, también FORWARD, BACK y LEFT requieren argumentos o valores de entrada.

Se puede volver a definir HEXAGONO para que también requiera argumentos de entrada. Para hacer esto

asignamos un nombre a la entrada y lo incluimos en la línea que contiene el título del procedimiento. El nombre de la entrada ha de ir siempre precedido por dos puntos. El nombre de la entrada y los dos puntos se utilizan en la definición del procedimiento, siempre que el valor de la entrada aparezca normalmente. Esto se entenderá mejor con un ejemplo.

Si al argumento de entrada para HEXAGONO le llamas LADO, al teclear EDIT "HEXAGONO estarás en modo editor. En la línea que contiene el título, añade ahora el nombre del argumento de entrada precedido por dos puntos, de forma que se lea:

```
TO HEXAGONO :LADO
```

En la siguiente línea suprime el 70 y teclea LADO en su lugar. La definición de hexágono queda ahora así:

```
TO HEXAGONO :LADO
REPEAT 6 [FORWARD :LADO
RIGHT 60]
END
```

Si ahora tecleas HEXAGONO, el LOGO te responderá con:

```
NOT ENOUGH INPUTS TO
HEXAGONO (faltan entradas
para HEXAGONO)
```

El trato que recibe HEXAGONO es el de cualquier otra primitiva del LOGO que requiere una entrada. Si tecleas HEXAGONO 20, la tortuga dibujará un hexágono de 20 unidades de lado. Con HEXAGONO 100 dibujará un hexágono de 100 unidades de lado, y así sucesivamente.

MODELO debería ser nuevamente escrito de la siguiente manera:

```
TO MODELO
REPEAT 12 [HEXAGONO 70
FORWARD 10 RIGHT 30]
END
```

Puedes modificar MODELO para tomar LADO como entrada.

```
TO MODELO :LADO
REPEAT 12 [HEXAGONO :LADO
FORWARD 10 RIGHT 30]
END
```

Utiliza este mismo principio en CIRCULO para hacer que la tortuga dibuje círculos de diferentes tamaños.

```
TO CIRCULO :MEDIDA
REPEAT 72 [FORWARD :MEDIDA
RIGHT 5
END
```

Realmente, la tortuga está dibujando un polígono de 72 lados en lugar de un verdadero círculo, aunque el resultado es una curva razonablemente regular. La tortuga dibuja los círculos bastante lentamente, pero puedes agilizar las cosas haciendo que la tortuga sea invisible; de esta forma, el LOGO no tendrá que dibujar a cada paso. Para hacer esto, utiliza HIDERTURTLE, que se puede escribir abreviadamente como HT.

Cuando quieras que la tortuga sea otra vez visible, utiliza SHOWTURTLE, abreviado ST. Ahora puedes modificar CIRCULO para que sea más rápido, así:

```
TO CIRCULO :MEDIDA
HIDETURTLE
REPEAT 72 [FORWARD :MEDIDA
RIGHT 5]
SHOWTURTLE
END
```

PONLO EN UN FICHERO

Una vez que un procedimiento ha sido definido, pasa a formar parte del vocabulario del LOGO, pero cuando apagues el ordenador, todas tus primitivas desaparecerán de la memoria. Con el LOGO puedes guardar tu trabajo, en disco o en cinta, utilizando SAVE seguido de ". Si se te olvidan las comillas, el LOGO no guardará nada y todos tus esfuerzos habrán sido en vano.

Al utilizar SAVE, el LOGO pone todo el contenido de tu "espacio de trabajo" —todos los procedimientos que hayas definido— en un disco o *cassette.* SAVE "TRABAJO guardaría todos los procedimientos que haya en la memoria del ordenador en un disco o cinta, creando un fichero llamado TRABAJO.

Después de cargar el LOGO, puedes recuperar el fichero del disco o de la cinta con LOAD (en algunos casos, con READ). LOAD "TRABAJO tomará todos los procedimientos que haya en el fichero llamado TRABAJO y los cargará en el área de trabajo del ordenador.

Si en la memoria del ordenador hay un procedimiento con un determinado nombre, y cargas otro procedimiento con el mismo nombre, el procedimiento recién cargado sustituirá al que ya estaba allí, el cual se perderá sin dejar el menor rastro.

Los ficheros que hay en el disco pueden borrarse del mismo con ERASFILE " y el nombre del fichero. A veces se utiliza ERS en vez de ERASEFILE. Así, ERASEFILE "TRABAJO borrará todos los procedimientos que estén el fichero TRABAJO del disco. No se puede utilizar ERASEFILE con un *cassette.* Para borrar un fichero de una cinta tienes que guardar otra área de trabajo en el sitio del fichero que quieres borrar.

El uso de ERASFILE no afecta a los procedimientos que están en la memoria del ordenador, sino sólo a los que ya han sido guardados en disco.

Como el LOGO guarda los procedimientos de la memoria sobre la base de "todo o nada", es conveniente ordenar el contenido de la misma antes de proceder a su almacenamiento en disco. Existen varias primitivas del LOGO relacionadas con la gestión del área de trabajo.

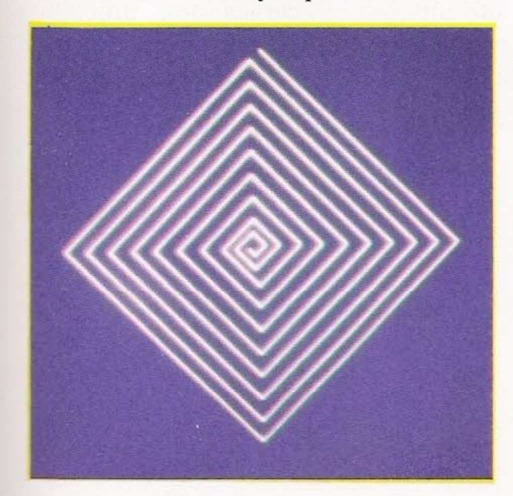
COMPROBACION DE LA MEMORIA

El área de trabajo de la memoria es una lista de nudos. Cada nudo consta de cinco bytes. Después de cargar el LOGO hay que manejar unos 3.000 nudos. Se utilizan al definir o cargar procedimientos. Para ver qué procedimientos hay en tu ordenador, utiliza POTS. Esto quiere decir *Print out titles* (escribe los títulos), escribiéndose los títulos de todos los procedi-

mientos que hay en el área de trabajo. Por ejemplo:

```
POTS
TO HEXAGONO :LADO
TO MODELO :LADO
TO FLOR
TO CIRCULO :MEDIDA
```

PO quiere decir imprimir *(Print Out)*. PO seguido del nombre de un procedimiento imprime la definición del mismo. Por ejemplo:



```
TO CIRCULO
TO CIRCULO :MEDIDA
HIDETURTLE
REPEAT 72 [FORWARD :MEDIDA
RIGHT 5]
SHOWTURTLE
END
```

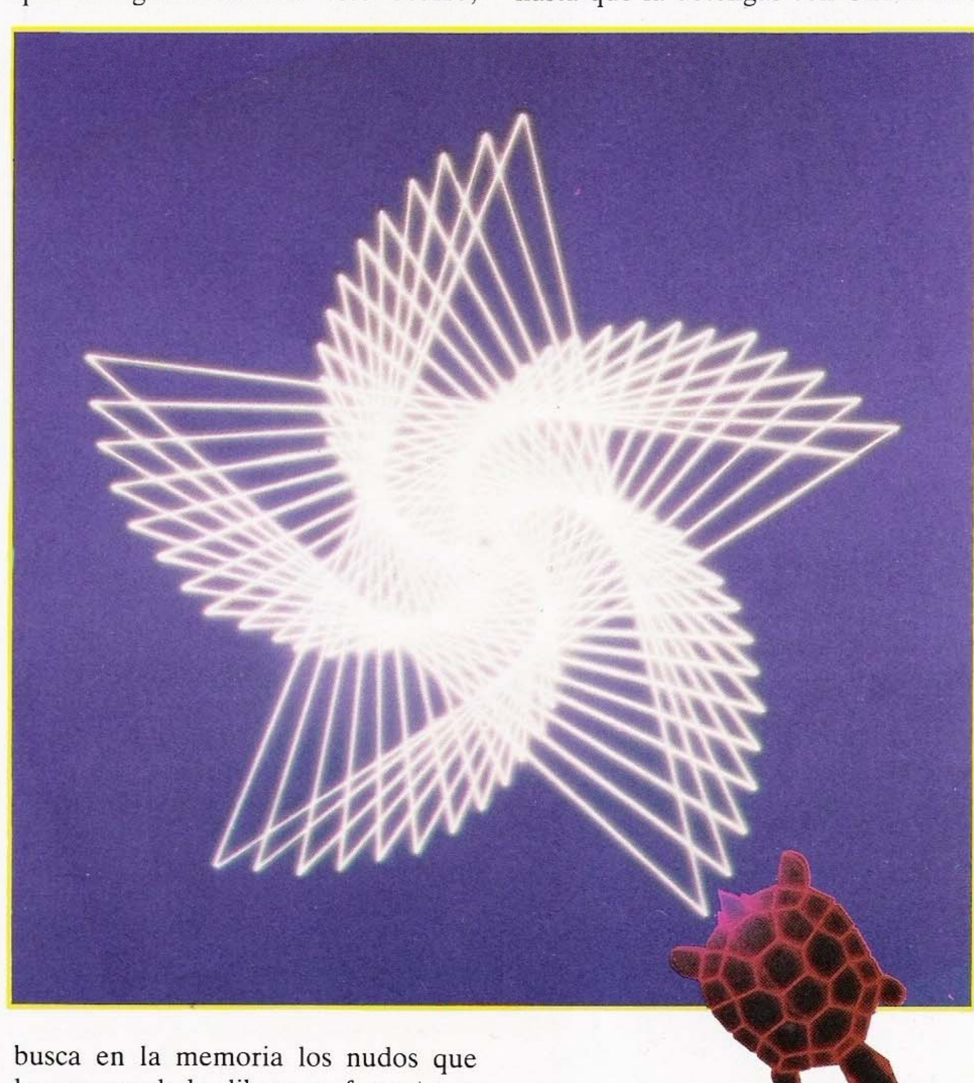
POALL significa imprimir todo. Imprime la definición de todos los procedimientos que hay en el área de trabajo. Después de ver qué procedimientos hay en la zona de trabajo puede ser que quieras borrar algunos de ellos. ERALL significa borrar todo *(Erase all)*, y borra todos los procedimientos que hay en la memoria.

Para borrar procedimientos aislados o en grupos utiliza ERASE. Para borrar un solo procedimiento, pon comillas delante del nombre de dicho procedimiento. Por ejemplo: ERASE "FLOR eliminará del área de trabajo el procedimiento FLOR.

Para borrar un grupo de procedimientos encierra sus nombres entre paréntesis cuadrados. Por ejemplo: ERASE [FLOR CIRCULO MODELO] borrará dichos procedimientos almacenados en la memoria del ordenador.

Con CATALOG obtendrás una lista de los ficheros almacenados en el disco que estás usando.

Al borrar procedimientos del área de trabajo vuelven a quedar nudos listos para ser utilizados de nuevo. No vuelven a añadirse a la lista de nudos libres del LOGO. Este continúa trabajando con su lista original hasta que se agota. Cuando esto ocurre, busca en la memoria los nudos que hayan quedado libres y forma con ellos una nueva lista. A esto se le llama "recogida de basura"» y a veces hace que el LOGO se pare durante un segundo o dos.



VUELTAS Y MAS VUELTAS

Volvamos de nuevo al tablero de dibujo. He aquí otra forma de dibujar el hexágono.

```
HEXAGONO 2 :LADO
END
```

Si lo pruebas, te quedarás confuso y aturdido. Dibuja un hexágono de un tamaño dado, igual que hacías HEXAGONO. Pero la tortuga se comporta de una forma muy diferente. En HEXAGONO, la tortuga se detenía cuando había terminado el dibujo. En HEXAGONO2, la tortuga sigue dando vueltas y más vueltas hasta que la detengas con **CTRL** más la tecla G. Esto sucede porque si, por ejemplo, tecleas HEXAGONO50, la tortuga ejecuta FORWARD50 RIGHT60 y entonces ha de hacer HEXAGONO2 50 una vez, y otra, y otra, hasta que la mandes parar.

Igual que un procedimiento puede llamar a otro procedimiento como parte de su definición, también puede llamarse a sí mismo como parte de sí

mismo. Este introvertido estado de cosas se llama recursión.

Un procedimiento puede tener más de una entrada. Aquí tienes un ejemplo sencillo:

```
TO RECTANGULO :LADO1 :LADO2
REPEAT 2 [FORWARD :LADO1
RIGHT 90 FORWARD :LADO2
RIGHT 90]
END
```

Así RECTANGULO20 40 dibujará un rectángulo de lados 20 y 40 unidades.

La entrada puede ser un ángulo o un lado.

```
TO MOVIMIENTO :LADO :ANGULO
REPEAT 60 [FORWARD :LADO
RIGHT :ANGULO]
END
```

MOVIMIENTO toma una entrada de su paso FORWARD y otro de la amplitud de su giro.

Un procedimiento muy corriente en LOGO, parecido a MOVIMIENTO, es POLY.

```
TO POLY :MEDIDA :ANGULO
FORWARD :MEDIDA
RIGHT :ANGULO
POLY :MEDIDA :ANGULO
END
```

POLY es un sencillo procedimiento recursivo que puede producir algunos dibujos atractivos.

Igual que HEXAGONO2, POLY pondrá a la tortuga a realizar un trabajo que no termina nunca. Tienes que detenerla interrumpiendo el programa.

Un desarrollo interesante de POLY, que va más allá de la mera repetición de las mismas instrucciones es ESPIRAL.

```
TO ESPIRAL :LADO :ANGULO
FORWARD :LADO
RIGHT :ANGULO
ESPIRAL :LADO+2 :ANGULO
END
```

Como el LOGO permite el uso de funciones matemáticas, en la cuarta línea se suma 2 al valor que tenía LADO.

Al teclear ESPIRAL 1 72 la tortuga ejecutará la siguiente secuencia:

```
FORWARD 1
RIGHT 72
FORWARD 3
RIGHT 72
FORWARD 5
RIGHT 72
FORWARD 7
RIGHT 72
FORWARD 9
```

y así sucesivamente. Esto va produciendo una espiral pentagonal. Cada lado de la espiral es dos pasos más largo que el lado anterior.

¡ALTO!

No hay manera de detener a la tortuga como no sea interrumpiendo el programa, pero puedes imponer una condición dentro del procedimiento, para que la tortuga se detenga cuando el lado alcance una determinada longitud.

STOP es una primitiva del LOGO que sólo puede utilizarse dentro de un procedimiento. No se puede utilizar de modo inmediato o a "alto nivel" cuando el programador tiene el control y el LOGO está esperando una instrucción. Detiene el procedimiento antes de que se llegue a la sentencia final y devuelve el control al usuario. El usuario podría ser otro procedimiento del LOGO. Si no hay otro procedimiento, el control vuelve al usuario de verdad a alto nivel. STOP solamente detiene el procedimiento en el que aparece.

También puedes imponer una condición con IF y THEN. En algunas versiones no se requiere THEN. IF examina alguna condición y decide si es cierta o no. Si es cierta, se pone en movimiento una determinada acción. Si la condición no es cierta, se pone en marcha otra acción diferente.

Con IF se prueba una condición dentro de un procedimiento; si no es cierta, se ejecuta la siguiente línea del programa. Si es cierta, THEN dice al LOGO lo que tiene que hacer.

En el siguiente procedimiento puedes ver en acción a STOP IF y THEN. En el procedimiento ESPIRAL, sustituye LADO por HEXAGONO, con lo que la tortuga dibujará una espiral de hexágonos cada vez más grandes. Los condicionales detendrán el procedimiento cuando los lados del hexágono alcancen las 20 unidades.

```
TO GIROHEX :LADO :ANGULO
IF :LADO>100 THEN STOP
HEXAGONO :LADO
RIGHT :ANGULO
GIROHEX :LADO+3 :ANGULO
END
```

El LOGO interpreta el signo > con el significado de "mayor que", del mismo modo que reconoció la función matemática de +.

GIROHEX 110 hace que la tortuga dibuje un hexágono de lado 1 unidad, gire 10 grados, dibuje un hexágono de lado 4 unidades, gire 10 grados, dibuje un hexágono de lado 7 unidades, y así sucesivamente, aumentando cada vez el lado del hexágono en 3 unidades. La segunda línea comprueba cada vez si los lados son mayores que 100. Si no es así, pasa a la línea siguiente y dibuja un nuevo hexágono. Cuando el lado se hace mayor que 100, entra en acción THEN. Le dice al LOGO que se pare (STOP) y el procedimiento se detiene.

En estos artículos hemos cubierto ya bastante material geométrico como para entretener durante un año a un chaval de diez años. La recursión hace posible obtener una amplia variedad de resultados a partir de programas sencillos. Con el editor de pantalla se pueden corregir los procedimientos y se estimula la experimentación. Los resultados de los programas son con frecuencia agradablemente sorprendentes. La tortuga convierte las instrucciones en imágenes y hace más fácil ver dónde están los errores de programación.

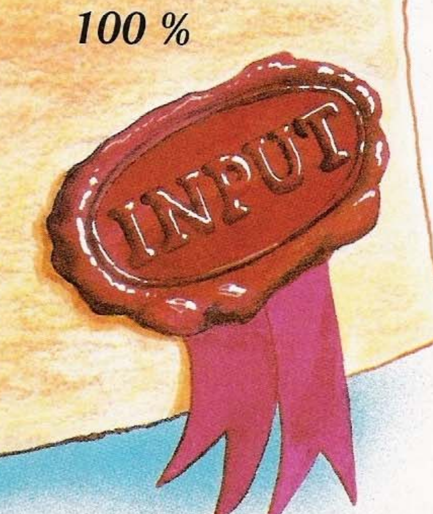
En la última parte de este artículo veremos los duendes del LOGO, el proceso de textos y de listas, y las posibilidades matemáticas.



LOS MEJORES DE INPUT COMMODORE

PUESTO	TITULO	PORCENTAJE
1.º	Gi Joe	20 %
2.º	Pistp-top II	16,8 %
3.º	Misión imposible	16,1 %
4.º	Panorama para matar	11,9 %
5.º	Entombed	10,4 %
6.º	The Exploding fist	7,6 %
7.º	The Dambusters	4,6 %
8.º	Bruce Lee	4,4 %
9.º	Beach Head II	4,2 %
10.º	Greemlins	4 %
		100 %

Octubre de 1985





PROTEGE TUS PROGRAMAS

- LO QUE PUEDES HACER
- AUTOEJECUCION
- DESACTIVACION DE SAVE Y LIST
- TRUCOS DIVERSOS

¿Quieres proteger tus técnicas especiales contra miradas indiscretas? ¿Tal vez desees añadir determinados toques profesionales a tu programa? A continuación te presentamos lo que el BASIC puede ofrecerte.



Siempre es una buena cosa proporcionarles a tus programas una apariencia profesional, una vez ajustadas todas las piezas. Siempre es posible mejorar su presentación, pero también puedes dotarle de una protección razonable para que no sean abiertos fácilmente.

La protección de los programas en BASIC recae en la inclusión de estrategias disuasorias, programas escritos total o mayoritariamente en código máquina, que pueden hacer uso de técnicas de protección más sofisticadas. Sin embargo, las empleadas para los programas en BASIC no pueden ser aplicados a los programas en código máquina.

Para prevenir la copia por parte de los «piratas» se pueden emplear todo tipo de trucos, aunque estos sólo deseen LISTar el programa por curiosidad. Cuantos más decidas integrar en tus programas, tanto mejor para ti. En otros programas no merce la pena el esfuerzo.

De una cosa puedes estar absolutamente seguro a este respecto: No hay forma de proteger a un programa que lo haga completamente seguro contra la copia. Mucha gente se toma lo de abrir programas como uno de los desafíos de la vida. Otros intentan entrar en el programa para examinarlo, aprender o simplemente modificar una rutina que le permita ajustarlo a sus necesidades.

PRIMEROS PASOS

Un método de proteger a los programas se centra en proporcionarles muchas trampas sencillas. Con ellas no se derrotará a quienes tengan algún conocimiento sobre la máquina, pero será una laboriosa tarea a realizar. Desafortunadamente, algo como esto puede convertirse en bastante tedioso para el escritor de programas, que deberá preocuparse de la protección al mismo tiempo que escribe el programa. El resultado podría traducirse en mayores complicaciones a la hora de depurar el programa.

Los cierres de este tipo hacen poco más que introducir cambios que hagan imposible el SAVE y LIST normales y dificultan algunos otros comandos relacionados con la posterior edición del programa.

CARGADORES

Una protección que resulta algo más ventajosa es disponer de un mecanismo de autoejecución del programa, una vez que se ha cargado. En los **Commodore,** la simple presión de las teclas SHIFT y RUN/STOP hace que el primer programa encontrado en la cinta sea ejecutado.

Esta orden se introduce, por regla general, en modo directo en el proceso normal de carga de un programa. Pero de un modo igualmente fácil puede ser reclamada desde otro programa almacenado delante del programa principal. Tal programa recibe el nombre de «cargador» y puede ser escrito en BASIC o en código máquina, dependiendo de la tarea específica que deba llevar a cabo. En su configuración más simple puede tomar la forma:

```
1ØØ LOAD «(NOMBRE DEL
 PROXIMO PROGRAMA)»
```

Ejecutando esta línea se cargará (LOAD) el programa cuyo nombre se ha escrito.

Por razones obvias, una sola línea como ésta no tiene mucho sentido. En realidad los cargadores pueden ser empleados para hacer mucho, mucho más. Son utilizados frecuentemente para portar programación suplementaria, responsable de cosas tales como visualizaciones de presentación a toda pantalla, avisos de *copyright,* carga e instrucciones de manejo, ajuste de variables y gran número de trucos destinados a la protección.

Entre ellos se incluyen algunas de las más potentes maneras de proteger un programa desarrollado en BASIC, logradas mediante la manipulación de los propios comandos del sistema. Pero el principal uso de los cargadores en este caso es el de programa de «puesta en marcha» que permite que el código máquina pueda ser ejecutado (RUN) desde el BASIC. Los cargadores son empleados en muchos ti-

PyR

¿Bajo qué circunstancias y en qué áreas de la memoria puedo almacenar en el Commodore 64 las rutinas para protección?

El lugar más obvio para ubicar tus rutinas es el área «escondida» de la RAM, situada entre las direcciones 49152 y 53247, pero es muy utilizada por cantidad de rutinas de carácter comercial. Sin embargo, existen algunos huecos, tales como las direcciones 251 a 255 de la Página Cero de la RAM. Un área ligeramente mayor es la comprendida entre 697 y 767.

pos de programas comerciales: muchos de ellos son grabados en forma de programas de partes múltiples, donde cada trozo es reclamado en respuesta al comando contenido en uno o más cargadores.

Con programas de partes múltiples los métodos de protección pueden recaer incluso en determinado nivel de interdependencia entre un fichero (uno de los programas) y otro. Lo que ocurre es que un fichero comprueba el valor de una dirección establecida por otro fichero. También podrías disponer de un programa que compruebe a un fichero de datos especial, grabado tras el programa principal. En cualquier caso, si desaparece algo, el sistema se bloquea o no se produce la ejecución.

En la mayoría de los casos, un programa de varias partes contendrá uno o más módulos escritos totalmente en código máquina.

Una ventaja final de los cargadores consiste en los tiempos de carga más rápidos posibles debido a la presencia del código máquina. Los datos de pantalla y caracteres pueden ser depositados directamente en la memoria, en lugar de utilizar las sentencias que incorporan los datos en BASIC y que sólo realizan su trabajo tras escribir RUN.

Todo lo anterior crea la impresión de acabado profesional que, incluso si ha sido escrito en BASIC, puede emplear RUN sin que parezca hacerlo.

AUTOEJECUCION

¿Cómo puede ser utilizado un cargador para ejecutar automáticamente un programa que le sigue? La única manera de conseguir que un programa haga esto en un **Commodore** consiste en disponer las instrucciones LOAD y RUN en el *buffer* del teclado, de tal forma que simulamos la presión de las teclas SHIFT y RUN/STOP, y devolviendo el control al BASIC en el mismo cargador.

El programa que sigue hace exactamente eso, manipulando una de las rutinas del sistema que incorpora el **Commodore.** Se llama **vector** a un par de bytes que dicen al sistema operativo que bit de su propio código es utilizado para un comando en particular. Más adelante veremos brevemente algo más sobre esto. (La versión para el **Vic-20** es para el modelo sin expansión de memoria).

Teclea para Commodore-64

```
10 N$="NOMBRE"
20 POKE 49189,LEN(N$)
30 FOR Z=1 TO LEN(N$)
40 POKE 49189+Z,ASC(MID$
   (N$,Z,1))
50 NEXT Z
60 FOR Z=679 TO 736
70 READ X
80 POKE Z,X
90 NEXT Z
100 FOR Z=49152 TO 49188
110 READ X
120 POKE Z,X
130 NEXT Z
200 POKE 770,167:POKE 771,2:
    SYS 49152
210 PRINT"BIEN":GOTO 210
220 DATA 169,47,133,0,169,
    55,133,1,32,138,255,169,
    1,141,32,208
230 DATA 169,48,141,119,2,
    169,76,141,120,2,169,
    207,141,121,2
240 DATA 169,13,141,122,2,
    169,82,141,123,2,169,
    213,141,124,2
250 DATA 169,13,141,125,2,
    169,7,133,198,108,0,160
260 DATA 162,1,160,1,169,1,
    32,186,255,162,38,160,
    192,173,37,192
270 DATA 32,189,255,169,
    167,133,251,169,2,133,
    252,162,5,160,3,169,251
280 DATA 32,216,255,96
```

Teclea para Vic-20

```
10 N$="NOMBRE"
20 POKE 7205,LEN(N$)
30 FOR Z=1 TO LEN(N$)
40 POKE 7205+Z,ASC(MID$
   (N$,Z,1))
50 NEXT Z
60 FOR Z=679 TO 723
70 READ X
80 POKE Z,X
90 NEXT Z
100 FOR Z=7168 TO 7204
110 READ X
120 POKE Z,X
130 NEXT Z
200 POKE 770,167:POKE 771,2:
    SYS 7168
210 PRINT"BIEN":GOTO 210
220 DATA 32,135,255
230 169,48,141,119,1,169,76,
    141,120,2,169,207,141,
    121,2
240 DATA 169,13,141,122,2,
    169,82,141,123,2,168,
    213,141,124,2
250 DATA 169,13,141,125,2,
    169,7,133,198,108,0,192
260 DATA 162,1,160,1,169,1,
    32,186,255,162,38,160,
    28,173,37,28
270 DATA 32,189,255,169,167,
    133,251,169,2,133,252,
    162,5,160,3,169,251
280 DATA 32,216,255,96
```

Almacena (SAVE) este programa de manera que puedas reclamarlo para su uso cuando sea necesario. Una vez hecho, vuelve a cargarlo (LOAD) escribiendo en la línea 10 el nombre del título que quieras autoejecutar. El segundo programa deberá almacenarse tras el cargador, si estás empleando cinta. Ejecuta el programa cargador (RUN), para situarlo en la me-

EL ZOCO



Cambio juego «Omega Race» para Vic 20 por cinta de juegos o de programas tecleados.

Julio de la Iglesia.
Zamora, 5, 2.º Decha.
Ciudad Rodrigo (Salamanca).

• • • • •

Intercambio programas para C-64, poseo 400. Mandar lista. Contesto a todas las cartas.

José M.ª Folch García.
Aparisi, 11, bajos 1.ª
Sabadell (Barcelona).

• • • • •

Vendo Vic-20 poco usado mas más muchos programas en cinta y revistas, todo por 15.000 pesetas.

Miguel Carrillo Sánchez.
Santander, 29.
Tel. (956) 65 51 25.

• • • • •

Deseo intercambiar programas del C-64. Mandar lista.

Pedro Luis Ríos Martínez.
P.º de la Cuba, 5. Te. (967) 21 21 38.
02001 Albacete.

• • • • •

Estoy interesado en el intercambio, compra y venta de juegos o programas del C-64.0/ y en formar un grupo de amigos commodoreros de la ciudad.

Antonio Jordán Masachs.
Catr. Antigua de Valencia, 2-4, 1.º, 1.ª
Tel. (93) 388 01 27.
Badalona (Barcelona).

• • • • •

moria. Posiciona la cinta y almacena el cargador, que en este caso incorpora el nombre del programa autoejecutable. Cuando aparezca el mensaje «CORRECTO» en la pantalla, presiona RUN/STOP y RESTORE para interrumpir el programa. Carga (LOAD) el segundo programa y PEEK a las direcciones 45 y 46 (direcciones que indican el comienzo del área destinado a las variables). Ahora escribe la siguiente línea al comienzo del segundo programa:

Teclea para C-64 y Vic-20

```
Ø POKE 45,X:POKE 46,Y
```

Siendo X e Y los valores que acabas de obtener PEEKeando las dos direcciones mencionadas. Pulsa RETURN y repite inmediatamente el PEEKeo, en modo directo, observando que los valores han cambiado. Retoca la línea 0 con los nuevos valores, repitiendo este ciclo hasta que los valores permanezcan constantes.

Ahora puede ser almacenado el segundo programa en la cinta. Pero, previamente, podrías querer incorporar otras medidas de seguridad, tales como los conocidos POKE que desactivan a determinados comandos.

LAS VARIABLES DEL SISTEMA

La autoejecución no es suficiente por sí misma. También debes proveer al programa de algunos medios que aseguren que no puede ser detenido, LISTado o amañado.

La manera de conseguirlo es llevar a cabo ajustes en la forma en que reacciona el sistema cuando se hace una llamada en particular, por ejemplo, a la subrutina LIST.

Cada ordenador viene completo con su sistema operativo y, en lo que nos concierne, un intérprete del BASIC. La mayoría de la información del sistema está contenida en una memoria que sólo puede ser leída, ROM, pero parte de ella es transferido a una memoria de acceso aleatorio, RAM, nada más conectarle la alimentación al ordenador. Esta es la información que puede ser alterada por el programador para modificar la forma en que trabaja el sistema, proporcionando medidas de seguridad bastante sofisticadas a incorporar en los programas.

En la medida en que van a trabajar con los modos de hacer del ordenador, necesitarás una documentada lista de las variables del sistema y las subrutinas. Si deseas profundizar aún más en el sistema, resulta esencial un buen mapa de memoria. El manual de referencia trae algo muy resumido.

Cuando conectamos el ordenador, parte de la información del sistema es volcada desde la ROM a la RAM, de manera que permite al sistema operativo alterar el valor de algunas de sus variables. En la medida en que la información está en la RAM es vulnerable a los cambios que el programador se tome la molestia de hacer.

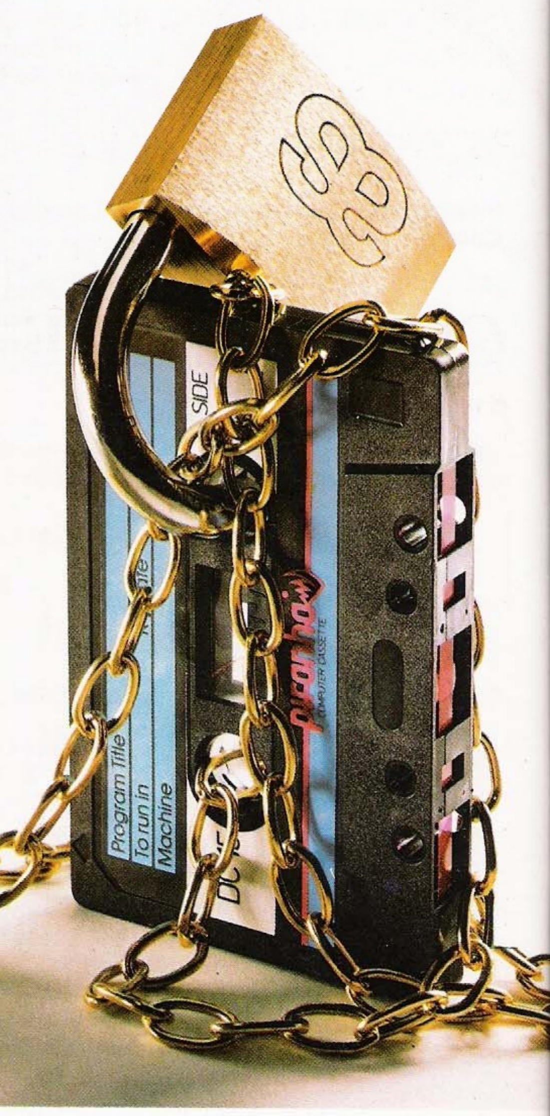
Un tipo especial de variable del sistema es el **puntero** o **vector.** Esto es normalmente un par de direcciones de memoria consecutivas que almacenan la dirección de una rutina específica. Si cambiamos la información sobre esta dirección, el sistema operativo se dirigirá al nuevo lugar indicado cuando sea requerida esa rutina. Todas las subrutinas del sistema están descritas, por supuesto, en código máquina y es por lo que la protección mediante el ajuste de las mismas es preferiblemente dejada en manos de nuestros programas escritos en código máquina también.

Aquellos **punteros** o **vectores,** de particular interés en lo que nos importa incluyen a los equivalentes del **vector** LIST, el **vector** de SAVE, los **vectores** de INTERRUPCION y el vector de RESET (Puesta a cero para inicializar el sistema) o WARM START (encendido en caliente, que equivale a inicializar el sistema sin desconectar el ordenador). Los dos últimos se sitúan fuera del objetivo de este artículo.

ESTUDIO DE CASOS

Resulta difícil, si no imposible, hacerse con una protección total contra los piratas —especialmente si tratas de evitar verte muy envuelto en el código máquina—. Los métodos que se pueden emplear en una máquina varían bastante entre modelos, debido a que el sistema operativo difiere de unos a otros. Pero, veamos unas cuantas ideas:

El programa de autoejecución y cualquier desactivación del teclado son fácilmente reconocibles, necesitándose un poco más de trabajo para



aumentar la dificultad. Esto es, utilizar la técnica de «borrado hacia atrás» para ocultar la presencia de los POKEs empleados para alterar la memoria y desactivar determinadas teclas. De hecho, esta técnica puede ser utilizada para enmascarar cualquier cosa dentro de una línea de programa —incluso los datos empleados para la comprobación de seguridad.

La técnica del borrado hacia atrás emplea mandos de borrado embutidos, contenidos entre comillas tras el inactivo REM en una línea de programa. Un ejemplo muestra con claridad como trabaja esta técnica, así que teclea lo siguiente, sin dejar espacios en blanco:

Teclea para C-64 y Vic-20

```
99 POKE 45,Ø:POKE
   46,2Ø:RUN:REM""SYS 4Ø96
```

Ahora desplaza el cursor (el cuadradito de la pantalla) hasta las segundas comillas y presiona SHIFT e INST/DEL hasta insertar 27 espacios. Tan pronto como hayas hecho esto, presiona solamente INST/DEL para introducir 27 borrados (DELetes). Estos símbolos toman la forma de Ts en inverso. Presiona ahora RETURN para introducir la línea. Vuelve a mover el cursor hacia arriba, para editar la línea (al comienzo del 99) y esta vez borra (DELete) las últimas comillas. Introduce la línea nuevamente con RETURN. Intenta ahora LISTarla. Si has seguido correctamente los pasos, todo lo que verás visualizado es:

```
1Ø SYS 4Ø96
```

Intenta editar la serie de «borradores» por medio de la inserción de espacios extra y añadiendo más símbolos de borrado.

Utilizando esta técnica es posible ocultar una línea completa, también puedes optar por hacerlo con sólo parte de una línea de programa.

Esta técnica se aplica solamente a la visualización en pantalla, un listado de impresora lo revelará todo.

Si recurres al programa cargador antes descrito, existe otro método de obtener protección adicional para el programa de autoRUN; incorporando una línea adicional de programa dentro del mismo, que PEEKee las direcciones utilizadas por el cargador. Esto significaría que el programa de autoejecución no podría ser cargado (LOAD) y ejecutado de otra manera que no sea con el cargador.

Añade la siguiente línea a tu segundo programa:

Teclea para Commodore-64

```
1 IF PEEK(679)<>169 OR
  PEEK(68Ø)<>47 THEN NEW
```

Teclea para Commodore-64

```
1 IF PEEK(679)<>32 OR
  PEEK(68Ø)<>135 THEN NEW
```

Incluye también los POKEs desactivadores del teclado adecuados, de tal forma que el programa no puede ser interrumpido mientras se ejecuta.

IMPRESORAS: LA ELECCION

- ¿NECESITAS UNA IMPRESORA?
- ELECCION DEL TIPO CORRECTO
- COMPRANDO EL PAPEL
- CONEXION DE LA IMPRESORA AL ORDENADOR

Si alguna vez has deseado tener una copia de tu programa o un volcado de pantalla de algún gráfico, es señal de que necesitas una impresora. Pero, ¿cómo averiguar qué tipo comprar?

La compra de una impresora es una forma de extender la utilidad de tu ordenador y de desarrollar tu habilidad de programación. Pero puede resultar una inversión muy costosa, y como las posibilidades y el margen de precios son tan variados como los ordenadores, solamente el micro-usuario mejor informado estará seguro de la necesidad de una impresora o de comprar la más adecuada.

La forma normal en que un ordenador presenta información al usuario es sobre una pantalla de televisión o sobre un monitor de vídeo. La impresora es una forma alternativa de presentar la misma información. Pero no pienses que ambos dispositivos tienen los mismos usos, ya que uno no es sustituto del otro. La principal misión de una impresora es entregar una copia o registro permanente *(hard copy)* de cualquier información que pueda aparecer en la pantalla.

Por ejemplo, hay muchos atractivos diseños gráficos —compu-arte, diríamos— que puedes imprimir, archivar e incluso construir, pero sin una impresora sólo existen de una forma efímera sobre la pantalla de un monitor. Si, por ejemplo, escribes un programa largo, es posible que quieras una copia del listado para trabajar sobre él fuera del ordenador. Si escribes o



Una sola columna de agujas sin punta golpea una cinta entintada para crear el conjunto de puntos que forman cada carácter, en una impresora de matriz de puntos.

Para que su COMMODORE trabaje

casa de software

PROCESADOR DE TEXTO

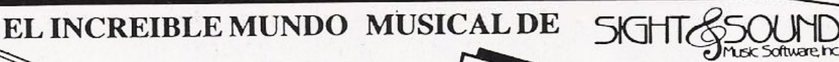
Versiones para Impresoras: IBM Compatibles (Centronics)
ASCII (Centronics o Bus serie COMMODORE)
SEIKOSHA SP 800
SEIKOSHA SP 1000 A

Programa en cartucho con posibilidad de grabación de documentos en cassette o diskette.
Esta primera versión ha sido desarrollada para la utilización de todas las posibilidades de la impresora SEIKOSHA SP-800, la cual permite el proceso de textos con una calidad de letra equiparable a la de las impresoras de margarita de precio mucho más elevado.
Caracteres castellanos y catalanes tanto en pantalla como en impresora. Posibilidad de utilizar todo el set de caracteres de la impresora. Márgenes, numeración de páginas, encabezamientos, pies de página, etc.
Los tres acentos y la diéresis se obtienen pulsando F1, F2, F3 o F4 y a continuación la vocal correspondiente como en una máquina de escribir convencional.
Posibilidad de cartas personalizadas (mail merge).

P.V.P. 14.900,- pts.

COMMODORE 64 casa de software PROCESADOR DE TEXTO

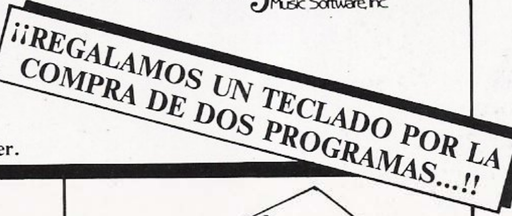
Este EJEMPLO ha sido escrito en letra de alta calidad mediante el cartucho ESCRITOR y la impresora SEIKOSHA SP 800.
En modo proporcional cada letra ocupa solamente el espacio necesario (la i ocupa menos espacio que la o).
La letra ITALICA permite personalizar sus escritos.
Puede escribir en negrita.
Subrayar cualquier frase.
Escribir en doble ancho.
Naturalmente dispone de los tres acentos (á à â) y la diéresis (ä) así como los caracteres ñ Ñ ç
Ç ¡ ¿ π µ ₧ ! * " # $ % & ' ( ) + - @ [ ] < > / = tanto en IMPRESORA como en PANTALLA.
Aproveche los Superíndices y los Subíndices.
Puede condensar cada tipo de letra: *ITALICA* ELITE y Doble ancho
¡Parece imposible!!! ¡¡¡Pero es verdad!!! ¡¡¡ES UN PRODUCTO DE CASA DE SOFTWARE!!!



EL INCREIBLE MUNDO MUSICAL DE SIGHT & SOUND Music Software, Inc

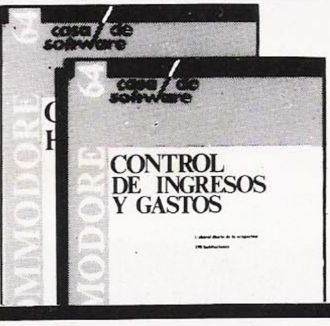


- Teclado musical.
- Music processor.
- Sound Odyssey.
- Music Vídeo Kit.
- Kawasaki Synthesizer.
- Kawasaki Rhythm Rocker.



Control de ingresos y gastos
Dispone de 99 cuentas y 250 apuntes por mes y disco (3.000 movimientos por año). Obtención de listados de cuentas y apuntes, balance mensual y anual. Programa ideal para Estimación Objetiva Singular. 9.900.– Pts.

Referencias bibliográficas
Creación de ficha para cada artículo, con referencia, tema y texto resumen de hasta 99 líneas. Búsqueda por los conceptos o códigos que precise. 9.900.– pts.



Cassette 7.500.– pts.

Otros Programas

MACHINE LIGHTNING
BASIC LIGHTNING
WHITE LIGHTNING
- **Contabilidad doméstica (cassette)**
- **Grabador de Eproms**
- **Grabador de Voz (Voice Master)**

NOVEDADES

- **Agenda telefónica. Marca el teléfono automáticamente. (disco)**
- **Etiquetas y recibos (cassette)**
- **Eureka (Matemáticas para B.U.P. y C.O.U.**

Administracion de fincas

PROPIEDAD VERTICAL:
30 inmuebles, 20 viviendas por inmueble, 30 propietarios, 5 poblaciones,
12 conceptos y antefirma en recibo.
Ingresos y gastos, listado de recibos, estado de cuentas y liquidaciones.

PROPIEDAD HORIZONTAL:
100 copropietarios y 504 asientos por comunidad. Conceptos de desglose, de recibo y desglose de gastos, control de impagados, etc. 35.000.– Pts.

Reservas Hotel: Control por quincenas del número de habitaciones reservadas durante todo el año. Listados y consultas. Tabla de valores para el cálculo de cambio de monedas extranjeras. 12.000.– Pts.

Ocupación Hotel: Control diario hasta 190 habitaciones. Tipo de pensión, núm. de personas, agencia, fecha entrada y salida. Listados. Búsqueda por todos los conceptos. 12.000.– Pts.

Adquiéralos en cualquier establecimiento autorizado o directamente a:
Casa de Software, s.a.
NUEVA DIRECCION:
C/. Taquígrafo Serra, 7, 5.ª planta.
Tel. 215 69 52
08029 BARCELONA

☐ Deseo recibir información de los siguientes programas:

____

☐ Deseo recibir contra reembolso los siguientes programas:

____

Nombre: ....
Dirección: ....
Población: ....

1021085

compras programas comerciales, necesitarás imprimir gráficos o cartas para ser incluidos en los informes. Y para el diseño asistido por ordenador necesitas un registro de formas u objetos al que poder hacer referencia desde el ordenador.

Todos los ordenadores domésticos tienen un teclado, que es semejante al de una máquina de escribir, de modo que puedes escribir y editar correspondencia, datos e incluso un libro, como sobre una máquina de escribir. Tu capacidad de manejar textos puede ir mucho más allá de lo que es meramente escribir a máquina si te sumerges en los dominios del proceso de textos, que te permiten manipular textos de múltiples formas. Con el *software* adecuado, tu micro tiene ya esta posibilidad, pero un bloque de texto muy cuidado, ya sea una carta o el contenido de un informe, es de poca utilidad si no se puede imprimir.

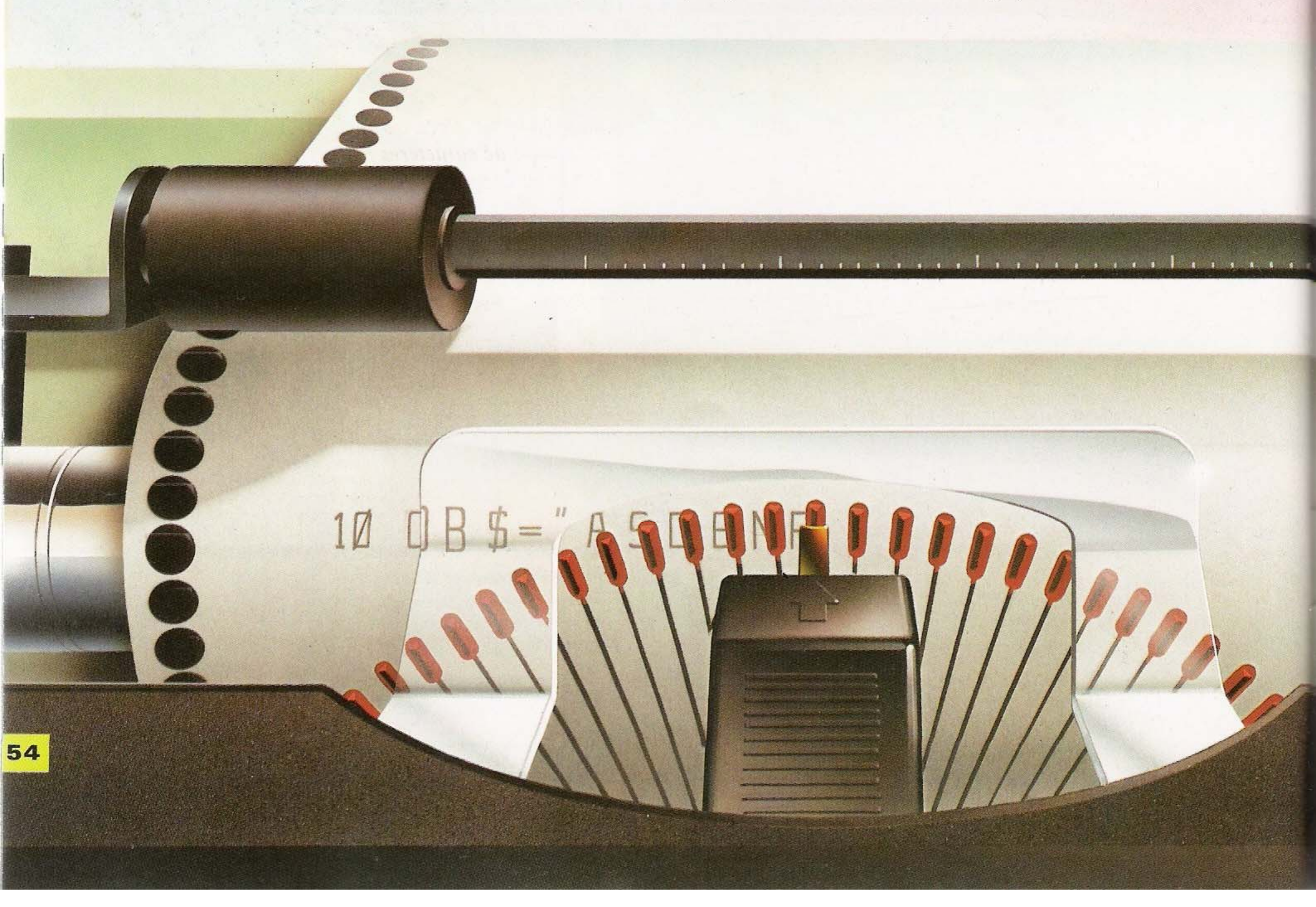
Y casi todos los usuarios encontrarán que una impresora es de gran utilidad como ayuda de programación. Además de proporcionar una copia de seguridad *(back-up)* de tu trabajo, con frecuencia es más fácil detectar errores sobre un papel que haciendo correr el texto del programa sobre la pantalla de un monitor. Un monitor tiene muchas ventajas, pero, en general, la gente prefiere la información impresa a la imagen electrónica, ya que le resulta más familiar.

Todas estas características pueden hacer que una impresora aparezca como una ayuda esencial para los usuarios del ordenador. Pero hay que decir que no todas las impresoras tienen todas estas posibilidades, y que las impresoras más versátiles tienen un precio prohibitivamente caro. Por ejemplo, no todas las impresoras pueden manejar gráficos, y sólo unas cuantas pueden imprimir en más de un color, normalmente (aunque no siempre) a un alto precio. Por esta razón es de la mayor importancia evaluar bien tus necesidades y elegir adecuadamente. El tipo que elijas depende de la aplicación que quieras darle. Para facilitar tu elección, empieza por la consideración de las características de los distintos sistemas disponibles.

TIPOS DE IMPRESORAS

Hay cuatro tipos de impresoras que podrían ser útiles para el usuario doméstico de un ordenador. Son la impresora por impacto de matriz de puntos, la de margarita, la térmica y la de plumilla.

El tipo más popular es la de matriz de puntos por impacto. Como todas las impresoras modernas, el carro portapapel permanece quieto, mientras que la cabeza impresora se mueve a lo ancho del papel. La cabeza impresora contiene una columna de agujas verticales muy agrupadas, cada una de las cuales tiene un solenoide asociado. Cuando un solenoide se activa, la aguja correspondiente recibe un fuerte empujón, golpeando el papel a través de una cinta impregnada de tinta, por lo que hará un punto sobre el papel. Actuando sobre la



combinación de agujas correcta, y haciendo avanzar la posición de la cabeza de impresión, se formará un carácter.

Los usuarios de **Commodore** pueden elegir entre dos impresoras de matriz de puntos específicas para sus máquinas, que se diferencian en sus prestaciones y en el precio. Su principal ventaja es que no necesitan interface y que imprimen los caracteres gráficos de ROM del **Commodore.**

Las velocidades de impresión de la impresora de matriz de puntos figuran entre las más rápidas disponibles, variando desde 80 caracteres por segundo (cps) a más de 400. Sin embargo, alguna de estas impresoras tiene la desventaja de que no presentan la «cola» de determinadas letras, tales como la «g» y la «q». Hacen que todo el carácter figure por encima de la línea, sin ninguna cola que asome por debajo. El poder presentar colas por debajo de la línea es deseable para tener claridad y una presentación atractiva, siendo una prestación disponible en las mejores impresoras de matriz de puntos.

Los primeros modelos de impresoras de matriz de puntos por impacto utilizaban una cabeza de impresión de siete agujas para producir caracteres en forma de matriz de 5 × 7 puntos, es decir, 5 columnas y 7 filas. Los modelos más recientes emplean nueve agujas para producir una matriz de 7 × 9 ó 9 × 9.

Incluso con la mejor impresora de matriz de puntos, los caracteres, aun siendo claramente legibles, aparecen como compuestos de puntos. Se adaptan muy bien para la mayoría del trabajo no profesional, y pueden imprimir dibujos a partir de la pantalla. Pero para una representación de calidad, se han hecho intentos de mejora de la claridad de impresión con la impresora de matriz de puntos.

Un método, llamado impresión multipaso, hace que la cabeza impresora imprima cada línea dos veces, una en una dirección, y una segunda vez en la dirección opuesta, todo ello sin hacer que el papel avance. En la segunda pasada, siempre hay, invariablemente, una pequeña falta de alineamiento entre las agujas impresoras y los caracteres ya impresos, por lo que las agujas tienden a imprimir en los espacios entre puntos. Se consiguen así efectivamente caracteres más marcados, menos punteados, pero, naturalmente, la velocidad de impresión queda reducida a la mitad.



Las impresoras de margarita son insuperables en cuanto a la calidad de la impresión. Para cambiar el conjunto dé caracteres no hay más que cambiar la propia margarita.

Otro método (que no implica una importante pérdida en la velocidad de impresión) utiliza múltiples columnas de agujas en la cabeza de impresión. Los caracteres se van formando en pasos sucesivos: se activa primero la columna de la derecha, cuando vuelve a su posición de equilibrio, se activa la columna que está más a su izquierda, etc.

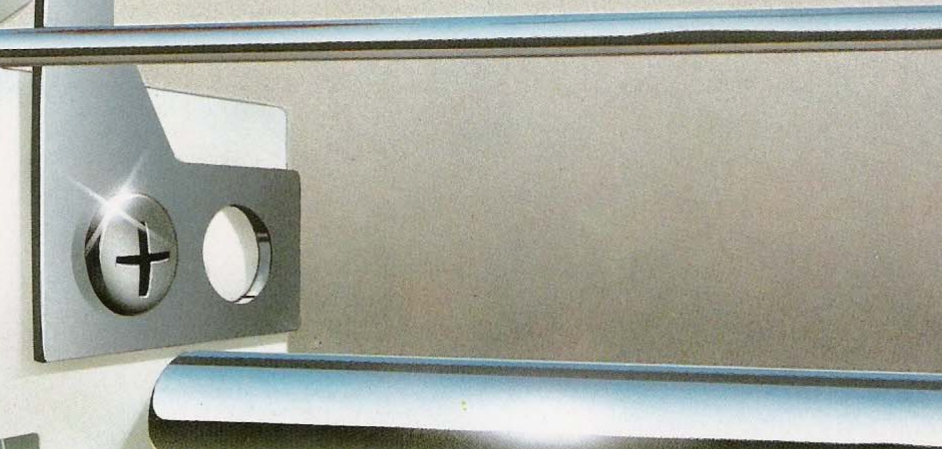
Aunque con estos métodos se consigue una mejora significativa, la calidad de la impresora de la matriz de puntos no es la mejor obtenible. Cuando se requiere una calidad elevada (como en la correspondencia formal), lo más adecuado es elegir una impresora de «fuente sólida». Fuente sólida quiere decir que las letras están formadas por tipos en relieve, que son una imagen especular exacta de la forma que se desea obtener, como ocurre con las letras de una máquina de escribir ordinaria.

IMPRESORAS DE FUENTE SOLIDA

Las primeras impresoras de fuente sólida que se usaron con los microordenadores eran adaptaciones de las máquinas de escribir eléctricas de bola. Aunque la calidad de impresión era excelente, la masa de la cabeza de impresión tipo pelota de golf (diseñada para una velocidad máxima de unos 15 caracteres por segundo, correspondiente a las mayores velocidades mecanográficas) retrasaba la impresión y la hacía poco fiable. En consecuencia, se fue desarrollando la impresora de margarita (conocida también como impresora de pétalos).

Las impresoras de margarita llevan un elemento impresor intercambiable que se parece a una margarita con los pétalos surgiendo radialmente desde un núcleo central. Cada pétalo lleva un carácter en la punta, que es golpeado por un martillo cuando ha llegado al sitio correcto. Como la margarita es extraíble, es muy sencillo cambiar de un tipo de letra a otro; para ello no hay más que cambiar las ruedas. La calidad de impresión es generalmente igual, o mejor, que la de una máquina de escribir y muy superior a la de una impresora de matriz de puntos por impacto, si bien las velocidades de impresión son más lentas que las de las impresoras matriciales. Son normales velocidades entre 8 y 70 caracteres por segundo.

Para mejorar la velocidad de impresión, muchas modernas impresoras de margarita son bidireccionales, lo que quiere decir que imprimen una línea de izquierda a derecha y la siguiente de derecha a izquierda. En otras palabras, aprovechan el retorno de carro, que antes era un tiempo perdido, para escribir una línea. Estas impresoras disponen también de circuitos que dicen a la cabeza impresora cuándo es más rápido volver al final de la línea más corta para imprimir la siguiente línea, o continuar hasta el final del papel e imprimir la línea siguiente en dirección contraria.



Las impresoras de plumilla encuentran gran aplicación en diseño con ayuda de ordenador, siendo muy útiles para generar una gran variedad de caracteres alfanuméricos.

Las impresoras de margarita permiten una serie de realces del texto. La negrita, el subrayado y las verdaderas colas son estándar en este tipo de impresoras. Se pueden combinar diferentes tipos de letra, cambiando las ruedas entre una primera y segunda pasada. Las modernas ruedas de margarita están hechas de plástico, por lo que son más ligeras. Los pétalos son elásticos, por lo que vuelven a su posición de partida después de cada golpe. Los caracteres están hechos de una resina dura. Golpean la cinta con unos bordes muy afilados, de forma que la calidad de la impresión es muy buena y la cinta dura mucho tiempo. Hay tres tamaños de caracteres: cuatro, cinco y seis caracteres por centímetro. El coste de sustitución de una rueda es del orden de una copia en *cassette* de un buen juego de aventuras.

IMPRESORAS SIN IMPACTO

Las impresoras de matriz de puntos sin impacto no son tan populares. Las más conocidas son las electrostásticas y las térmicas. Las impresoras térmicas requieren utilizar un papel con un recubrimiento químico sensible al calor. El papel es caro, pero no hace falta cinta. La cabeza de impresión contiene una matriz de pequeños elementos calentadores. Una vez que se ha seleccionado la figura que ha de ser impresa, los elementos que la componen se calientan a una temperatura entre 100 y 150° C, oscureciendo los tintes termosensibles del papel.

Las impresoras electrostáticas requieren una aportación de *toner* líquido (normalmente partículas de carbón suspendidas en una solución de isoparafina). Los caracteres se forman cargando el papel cubierto y haciéndolo pasar por una solución de toner. Las partículas negras del *toner* se adhieren a las zonas eléctricamente cargadas del papel. El exceso de toner es eliminado y el papel se calienta para el fijado.

Las impresoras térmicas y electrostáticas son más fiables que las de matriz de puntos por impacto (porque tienen menos partes mecánicas) pero tienen una resolución mucho más baja.

Uno de los desarrollos recientes más interesantes es un registrador/impresora de plumilla de bajo costo. Sirve para dibujar (con plumillas con su propio depósito de tinta) caracteres preprogramados, si bien fue concebido específicamente para hacer dibujos. En los modelos de más bajo precio, se utilizan plumillas tipo bolígrafo para dibujar sobre un papel corriente.

Típicamente, la cabeza escritora contiene hasta cuatro plumillas, por lo que se pueden hacer gráficos multicolores. El conjunto de caracteres alfanuméricos normales resulta nítido y fácilmente legible, pero puedes programar tus propios caracteres utilizando las posibilidades de gráficos de tu ordenador. Las velocidades de impresión con el conjunto de caracteres normales son del orden de unos 12 caracteres por segundo. Un juego de cuatro plumillas de repuesto viene a costar aproximadamente lo mismo que un par de *cassettes* de datos.

CONSIDERACIONES SOBRE EL PAPEL

Cuando se elige una impresora lo más importante es saber qué aspectos quieres que tenga lo impreso. Esto se aplica no sólo al tipo de letra, sino también al tamaño del papel y a su apariencia. Además de los papeles con recubrimientos requeridos por las impresoras electrostáticas y térmicas, hay una amplia variedad de anchuras, que va desde 10 a 40 cm. Algunas impresoras pueden utilizar varios tipos y tamaños de papel; otras sólo uno. El papel se puede comprar en hojas sueltas (como el papel de escribir normal), en rollos o plegado. El papel plegado es el que normalmente se asocia con las copias de ordenador, lleva en cada borde una banda agujereada y se suministra como una hoja continua, con perforaciones en cada «página» para poder separarlo en hojas y cortarlo a la longitud conveniente.

Cada tipo de papel requiere su propio mecanismo de arrastre. Para el papel plegado se utiliza tracción por medio de ruedas dentadas. Los agujeros de los bordes del papel van engranando con los salientes de las ruedas de arrastre, que van girando para ir haciendo pasar el papel ante la cabeza impresora. Cuando se utiliza una impresora de impacto, se pueden sacar copias múltiples intercalando hojas de papel carbón.

Cuando el papel viene en hojas sueltas o en rollo, se requiere un mecanismo de arrastre por friccioń (como el de las máquinas de escribir). El papel queda atrapado entre dos rodillos de presión. El papel en rollos se alimenta desde un eje; para las hojas sueltas se requiere una bandeja de entrada independiente. Para la impresión de correspondencia de gran calidad de presentación, lo mejor es sin duda utilizar hojas sueltas; el papel plegado y en rollo están más indicados para listados de programas y gráficos.

INTERFACE

Independientemente del tipo de impresora que decidas comprar, el punto más importante a considerar es si es compatible, o puede hacerse compatible, con tu ordenador. El ordenador sólo puede comunicarse con la impresora si ambas máquinas están adecuadamente interconectadas.

Un *interface* consiste esencialmente en el *hardware* (junto con el posible *software* requerido para controlarlo) que hace posible conectar dos sistemas. Algunos sistemas se conectan simplemente con un enchufe, porque ya se han hecho compatibles durante su fabricación o porque se venden con *interfaces.* Otros se venden sin nada de *interface,* o con sólo una parte de *interface,* por ejemplo, sólo con el *hardware.*

No esperes a haber comprado la impresora para averiguar las exigencias del *interface,* porque te puedes encontrar con que el coste del *interface* asciende a un tercio del coste de la máquina.

Si ambas máquinas proceden del mismo fabricante, lo normal es que ya sean compatibles, o puedan serlo añadiéndoles un *interface.* Y los fabricantes independientes de impresoras intentan hacer sus productos compatibles con el mayor número de ordenadores posible, normalmente suministrando los *interfaces* adecuados. Por eso, aunque el *interface* es crucial, no tiene por qué ser un problema, si te aseguras bien antes de comprar.

Se pueden comprar módulos de conversión de *interface* para algunas máquinas, pero también puede ocurrir que el *software* que quieras utilizar —especialmente algunos programas procesadores de textos— requiera su *interface* particular.

Cuando hayas decidido el tipo de impresora que vas a comprar, tienes que tener una idea del precio, *interface* incluido. Naturalmente, puede que el precio haya sido desde el principio el factor decisivo. Ten cuidado con algunas máquinas que se venden sin cable de interconexión, que puede llegar a costarte el doble que muchos juegos de disco de bajo precio.

Tanto si eliges una impresora que cueste tanto como tu ordenador, como si cuesta diez veces más, te recomendamos que examines el paquete de *software* y la impresora que quieres ver trabajando junto con tu ordenador, antes de quedarte solo.

LOS MEJORES DE INPUT

*Hemos pensado que es interesante disponer de un **ranking** que ponga en claro, mes a mes, cuáles son los programas preferidos de nuestros lectores. Para ello, es obligado preguntaros directamente y tener así el mejor termómetro para conocer vuestras preferencias. Podríamos haber recurrido a otros métodos de muestreo, pero éste no deja lugar a dudas.*

*El resultado de las votaciones será publicado en cada número de **INPUT**.*

Entre los votantes sortearemos 10 cintas de los títulos que pidáis en vuestros cupones.

***Nota:** No es preciso que cortéis la revista, una copia hecha a máquina o una simple fotocopia sirven.*

Enviad vuestros votos a:

LOS MEJORES DE INPUT
Alberto Alcocer, 46-4.° B
28016 Madrid

ELIGE TUS PROGRAMAS

Primer título elegido

....

Segundo título elegido

....

Tercer título elegido

....

Programa que te gustaría conseguir

....

¿Qué ordenador tienes? ....

Nombre |_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|

Apellidos |_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|

Dirección |_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|

|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|

Teléfono de contacto ....

INPUT COMMODORE

FRANKIE VA A HOLLYWOOD

Este es un juego de estrategia y habilidad. Nuestro protagonista está en un gran aprieto: ¡no tiene personalidad! Eres tú, con tu habilidad y la ayuda de **Frankie,** el que tienes que darle los puntos necesarios para ir proporcionándole personalidad mediante investigaciones en las pantallas por donde vas a pasar.
Durante la primera parte encuentras cuatro casas similares (¡ten cuidado, porque parecen iguales!). Cada una de ellas tiene cuatro puertas de entrada. Una vez traspasada una de ellas, te encuentras en un vestíbulo con otras dos o tres puertas: una es por la que has entrado; otra, es el salón, y la tercera (que no aparece siempre) es la cocina. Estas cuatro casas —por las que irás pasando— significan el amor, el sexo, la muerte y el odio. Están representadas por símbolos en la parte inferior derecha de la pantalla, y, sobre cada uno de ellos, hay una columna cuya altura indica los descubrimientos que vas haciendo en cada casa.
Tu misión es investigar aquellos objetos que encuentres a tu paso, porque dan puntos para incrementar la personalidad. La investigación se realiza situándose delante del objeto y pulsando «fuego» con el *joystick* en posición «hacia arriba». Verás que tu personaje levanta el brazo y señala con un dedo el objeto en cuestión. Aparecerá en la pantalla una ampliación del objeto que estudias (armarios de cocina, cajones del salón...) con cosas que puedes llevarte como ayuda, entre las que encontrarás desde armaduras, vídeos, pescado..., hasta pistolas y corazones. Dispones de un inventario al que puedes acceder pulsando «fuego» y moviendo el *joystick* hacia atrás, obteniendo una pantalla con los objetos que llevas contigo. Para coger un objeto, lo único que tienes que hacer es señalar con la mano aquello que deseas o bien la salida. El objeto que hayas elegido aparecerá reflejado en tu inventario, que sólo podrá contener ocho objetos nada más. Cada objeto tiene su utilidad (¡no hemos encontrado todas!). El vídeo te sirve para ponerlo en la televisión que encontrarás en algunos salones cuando las investigues, entonces saldrá una pequeña pantalla con imágenes del vídeo que te servirán de ayuda.
Procura no liarte mucho con las casas, ya que éstas se comunican entre sí por las puertas traseras de las cocinas y es preciso que sepas en cuál de ellas te encuentras. Por eso, es recomendable que, conforme vayas avanzando por casas y habitaciones, hagas un plano de cómo puedes acceder de una casa a otra.



Puertas y más puertas.



Lo principal es catalogar los objetos.



¡Lo que faltaba!, ¡un cadáver!



Vista exterior de una de las casas.

DATOS GENERALES	CALIFICACION
TITULO: Frankie goes to Hollywood	GRAFICOS: 4,5 sobre 5
FABRICANTE: Ocean	COLOR: 4 sobre 5
ORDENADOR: Commodore	PRESENTACION: 4,5 sobre 5
MEMORIA: 64 K	INTERES: 4,5 sobre 5
CLASE DE PROGRAMA: Aventura doméstica	REALISMO: 4 sobre 5



METODOLOGIA DE LA PROGRAMACION

El primer libro recomendado para escuelas de enseñanza de informática y para aquellas personas que quieren aprender la programación. Cubre las especificaciones del Ministerio de Educación y Ciencia para Estudios de Informática. Realizado por un alto mando del ejército Español, un Dr. Ingeniero y Diplomado en Informática y profesor de la UNED y por un oficial técnico especialista en informática de gestión. Utilizado en todos los institutos politécnicos del ejército español. Es un seguro candidato a ediciones en lengua inglesa, alemana y francesa. Es el primer libro que introduce a la lógica del ordenador. Es un elemento de base que sirve como introducción para la programación en cualquier otro lenguaje. No se requieren conocimientos de programación ni siquiera de informática. Abarca desde los métodos de programación clásicos a los más modernos. **Precio venta 2.200 ptas.**

MANUAL ESCOLAR PARA SU COMMODORE 64

Este libro, escrito especialmente para escolares de grado medio y superior, contiene muchos interesantes programas de aprendizaje para solucionar problemas, descritos detalladamente y de manera fácilmente comprensible. Facilitan un aprendizaje intensivo y ameno, con, entre otros, los siguientes temas: Teorema de pitágoras, progresiones geométricas, palanca mecánica, crecimiento exponencial, verbos irregulares, ecuaciones de segundo grado, movimientos de péndulo, formación de moléculas, aprendizaje de vocablos, cálculo de interés y su capitalización. Una corta repetición de los elementos BASIC más importantes y una introducción a los rasgos esenciales del análisis de problemas, entre otros, completan el conjunto. **Precio de venta 2.800 ptas.**

64 EN EL CAMPO DE LA TECNICA Y LA CIENCIA

Ofrece un campo fascinante y amplio de problemáticas científicas. Para esto el libro contiene muchos listados interesantes: Análisis de Fornier y síntesis, análisis de redes, exactitud de cálculo, formateado de números, cálculo del valor PH, sistemas de ecuaciones diferenciales, modelo ladrón presa, cálculo de probabilidad, medición de tiempo, integración, etc. **Precio venta 2.800 ptas.**

ROBOTICA PARA SU COMMODORE 64

En el libro de los robots se muestran las asombrosas posibilidades que ofrece el CBM 64, para el control y la programación, presentadas con numerosas ilustraciones e intuitivos ejemplos. El punto principal: Cómo puede construirse uno mismo un robot sin grandes gastos. Además, un resumen del desarrollo histórico del robot y una amplia introducción a los fundamentos cibernéticos. Gobierno del motor, el modelo de simulación, interruptor de pantalla, el Port-Usuario cómodo del modelo de simulación, Sensor de infrarrojos, concepto básico de un robot, realimentación unidad cibernética, Brazo prensor, Oir y ver. **Precio venta 2.800 ptas.**

TODO SOBRE EL FLOPPY 1541

La obra Standard del floppy 1541, todo sobre la programación en disquettes desde los principiantes a los profesionales, además de las informaciones fundamentales para el DOS, los comandos de sistema y mensajes de error, hay varios capítulos para la administración práctica de ficheros con el FLOPPY, amplio y documentado Listado del Dos. Además un filón de los más diversos programas y rutinas auxiliares, que hacen del libro una lectura obligada para los usuarios del Floppy. **Precio venta 3.200 ptas.**

MANTENIMIENTO Y REPARACION DEL FLOPPY 1541

Saberse apañar uno mismo, ahorra tiempo, molestias y dinero, precisamente problemas como el ajuste del floppy o reparaciones de la platina se pueden arreglar a menudo con medios sencillos. Instrucciones para eliminar la mayoría de perturbaciones, listas de piezas de recambio y una introducción a la mecánica y a la electrónica de la unidad de disco, hay también indicaciones exactas sobre herramientas y material de trabajo. Este libro hay que considerarlo en todos sus aspectos como efectivo y barato. **Precio venta 2.800**

EL MANUAL DEL CASSETTE

Un excelente libro, que le mostrará todas las posibilidades que le ofrece su grabadora de casettes. Describe detalladamente, y de forma comprensible, todo sobre el Datassette y la grabación en cassette. Con verdaderos programas fuera de serie: Autostart, Catálogo (¡busca y carga automáticamente!), backup de y a disco, SAVE de áreas de memoria, y lo más sorprendente: un nuevo sistema operativo de cassette con el 10-20 veces más rápido FastTape. Además otras indicaciones y programas de utilidad (ajuste de cabezales, altavoz de control). **190 pág. P.V.P. 1.600,- ptas.**

EL DICCIONARIO PARA SU COMMODORE 64

Este es el libro que buscaba: una enciclopedia exhaustiva del C 64 y su programación, un diccionario general de micros que contiene toda la terminología informática de la A a la Z y un diccionario técnico con traducciones de los términos ingleses de más importancia - los DICCIONARIOS DATA BECKER prácticamente son tres libros en uno. La increíble cantidad de información que contienen, no sólo los convierte en enciclopedias altamente competente, sino también en herramientas indispensables para el trabajo. El DICCIONARIO DATA BECKER se edita en versión especial para APPLE II, COMMODORE 64 e IBM PC. **350 pág. P.V.P. 2.800,- ptas.**

LENGUAJE MAQUINA PARA COMMODORE 64

¡Por fin una introducción al código máquina fácilmente comprensible! Estructura y funcionamiento del procesador 6510, introducción y ejecución de programas en lenguaje máquina, manejo del ensamblador, y un simulador de paso a paso escrito en BASIC **1984, 201 pág. P.V.P. 2.200,- ptas.**

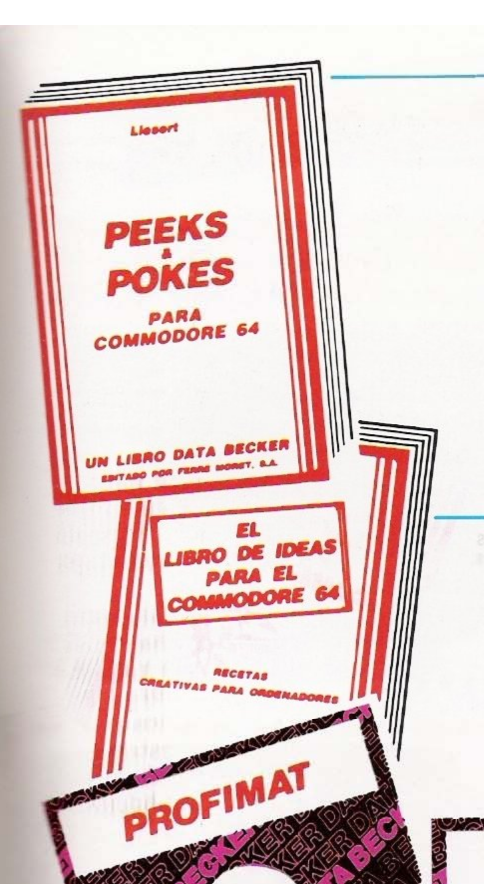
64 INTERNO

Con más de 60.000 ejemplares vendidos, ésta es la obra estándar para el COMMODORE 64. Todo sobre la tecnología, el sistema operativo y la programación avanzada del C-64. Con listado completo y exhaustivo de la ROM, circuitos originales documentados y muchos programas. ¡Conozca su C-64 a fondo! **1984, 352 pág. P.V.P. 3.800,- ptas.**

64 CONSEJOS Y TRUCOS

CONSEJOS Y TRUCOS, con más de 70.000 ejemplares vendidos en Alemania, es uno de los libros más vendidos de DATA BECKER. Es una colección muy interesante de ideas para la programación del Commodore 64, de POKEs y útiles rutinas e interesantes programas. Todos los programas en lenguaje máquina con programas cargadores en Basic. **1984, 364 pág. P.V.P. 2.800,- ptas.**





PEEKS y POKES

Con importantes comandos PEEK y POKE se pueden hacer tambien desde el Basic muchas cosas, para las que se necesitarian normalmente complejas rutinas en lenguaje maquina. Este libro explica de manera sencilla el manejo de PEEKs y POKEs. Con una enorme cantidad de POKEs importantes y su posible aplicacion. Para ello se explica perfectamente la estructura del Commodore 64: Sistema operativo, interpretador, pagina cero, apuntadores y stacks, generador de caracteres, registros de sprites, programación de interfaces, desactivación del interrupt. Ademas una introduccion al lenguaje maquina. Muchos programas ejemplo. **177 pág. P.V.P. 1.600,- ptas.**

MSX LENGUAJE MAQUINA

El libro del Lenguaje Máquina para el MSX está creado para todos aquellos a quienes el BASIC se les ha quedado pequeño en cuanto a rendimiento y velocidad. Desde las bases para la programación en Lenguaje Máquina, pasando por el método de trabajo del Procesator Z-80 y una exacta descripción de sus órdenes, hasta la utilización de rutinas del sistema, todo ello ha sido explicado en detalle e ilustrado con múltiples ejemplos en este libro.
El libro contiene, además, como programas de aplicación, un ensamblador, un desensamblador y un monitor. ¡Así es cómo se facilita el acceso al Lenguaje Máquina! **305 pàginas. 2.200,- ptas.**

Dullin · Strassenburg

MSX Lenguaje Máquina

UN LIBRO DATA BECKER
EDITADO POR FERRE MORET, S.A.

EL LIBRO DE IDEAS DEL COMMODORE 64

Casi todo lo que se puede hacer con el Commodore 64, está descrito detalladamente en este libro. Su lectura no es tan sólo tan apasionante como la de una novela, sino que contiene, además de listados de utiles programas, sobre todo muchas, muchas aplicaciones realizables en el C64. Se ha valorado especialmente, que el libro sea de fácil comprensión para los no iniciados. En parte hay listados de programas listos para ser tecleados, siempre que ha sido posible condensar «recetas» en una o dos páginas. Si hasta el momento no sabia que hacer con su Commodore 64, ¡después de leer este libro lo sabrá seguro! **1984, más de 200 páginas, P.V.P. 1.600,- ptas.**

Software

PROFIMAT

TEXTOMAT



PROFIMAT 5.850 ptas.

Quien quiere acceder a la vida interior del computador, necesita disponer de herramientas especiales. Aquí es donde PROFIMAT entra en accion, ofreciendo un confortable monitor de codigo máquina asi como un macroensamblador.

ADA — curso de entranamiento 11.700 ptas.

Este lenguaje de programacion del futuro, el cual al igual que COBOL fue encargado por el pentagono, ahora puede ser aprendido por el usuario del C-64, mediante el curso de entrenamiento de DATA BECKER. El curso de entrenamiento de ADA ademas contiene un compilador que ofrece un exhaustivo SUBSET, y los elementos principales de este lenguaje.

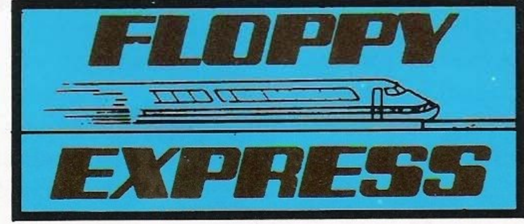
TEXTOMAT 5.850 ptas.

Todo el programa se estructura en forma de menú, para mayor facilidad de manejo. Puede seleccionarse a voluntad el juego de caracteres americano, o el juego de caracteres castellano. ¡Esta adaptación tambien vale para la impresora, y posibilita una adaptación individual a cualquier problema!

ELECTROMAT 4.100 ptas.

Con ELECTROMAT, la creacion de esquemas de circuitos electronicos se convierte en una diversion - todos los esquemas pueden editarse directamente en pantalla.

Hardware



17.600 ptas.

Le da marcha al Floppy VC. 1541

¡La velocidad de transmisión de datos aumenta 25 veces!

PLATINE 64

Hasta ahora los sistemas de desarrollo de placas de circuito impreso* (platinas) sólo eran accesibles con mucho dinero y a base de computadoras caras.

¿QUIEN NECESITA PLATINE 64?

- Técnicos de desarrollo electrónico y pequeñas empresas que hasta ahora no podían costearse un sistema de desarrollo de placas. Placas de circuitos impresos.
- Departamentos especializados de empresas mayores que buscan más independencia, hallándola en un sistema rentable y autónomo.
- Aficionados serios a la electrónica. Por sus características excepcionales de rendimiento.

En una de las casas (¡adivina cuál!), al entrar en un salón, encontrarás un **muerto.** Ahora tu misión, además de aumentar la personalidad, es descubrir al terrible asesino mediante las pistas que te irán dando, conforme vayas pasando por las ditintas habitaciones. Estas pistas te van descubriendo datos sobre todo los posibles sospechosos del crimen, y también del asesino. Por eliminación, deberás quedarte sólo con aquel que puede cumplir las características del asesino. Cuando estés seguro, debes volver a la habitación donde encontraste al muerto, y aquí te aparecerá una lista con todos los personajes: tienes que señalar al asesino. Procura estar bien seguro, pues si te equivocas de individuo descenderá considerablemente el nivel de personalidad.
En uno de los muchos salones encontrarás una puerta, por la cual no puedes pasar, porque está cerrada. Necesitas una llave para abrirla. La llave la podrás encontrar en una de las habitaciones. ¡Búscala porque entrarás en una nueva etapa del juego!
Una vez dentro, existe un laberinto lleno de pasillos y con muchas salas diferentes. Pero esta vez no vamos a ser nosotros quien os descubra las pantallas restantes: esperamos que con estas indicaciones y vuestra habilidad consigáis llegar al final. ¡No os lo vamos a dar todo hecho...!

****

A GOLPE DE PEDAL

Activision ha lanzado este verano tres juegos nuevos: el **Tour de Francia, Fireworks** y **Great American Cars Country Road Race** (GACCRR); de los tres, vamos a comentar este mes el **Tour.**
Tal y como sucede en la realidad este *tour* consta de dieciséis etapas, que podrás recorrer de dos maneras: en modo de competición y en modo de entrenamiento. Tras la elección aparece un mapa del país, con el trazado del *Tour*, indicando las ciudades por las que pasa. Eligiendo el modo de entrenamiento habrás de decidir en este mapa, las diferentes etapas que quieres recorrer.
Al igual que en **Summer Games,** tienes que elegir nombre del jugador y bandera del país que representa. Puedes tener hasta seis jugadores, aunque no podrán jugar al mismo tiempo.
Con la primera opción (competición), empezarás la carrera en PARIS. En la segunda podrás elegir las etapas que quieras, indicando con un círculo el principio de etapa. Las elecciones no se limitan únicamente a esto, sino que también tienes opción de elegir el tipo de bicicleta (tecla f.1), así como el *maillot* (tecla f.3). Dispones de dos velocidades, que regulas con el *joystick:* pulsando hacia arriba el manillar (¡perdón!, la palanca) alcanzas la velocidad máxima que aparece indicada con la letra H y verás junto al contador de tiempo. El movimiento inverso reduce la velocidad (en este caso la letra correspondiente es la B), manteniendo la palanca pulsada hacia ti frenas. Hay que mencionar una característica particular de este programa: el color del contador te indica si te hallas por encima (color azul) o por debajo (color rojo) del tiempo medio de cada etapa.
En cuanto a su acabado, el juego está muy bien planteado. Los gráficos están muy cuidados. Tal como puedes ver en la fotografía, concurre numeroso público en las etapas y el recorrido está ambientado con anuncios publicitarios, entre los que hay que destacar uno sorprendente (Sinclair). Otra característica a destacar es la música, que cambia durante las distintas etapas.
Tienes que practicar durante algún tiempo para poder dominar el



Las etapas, sobre el mapa de Francia.



Los gráficos de este programa son francamente agradables.

DATOS GENERALES	CALIFICACION
TITULO: Le Tour de France	GRAFICOS: 4,5 sobre 5
FABRICANTE: Activision	COLOR: 4,5 sobre 5
ORDENADOR: Commodore	PRESENTACION: 4,5 sobre 5
MEMORIA: 64 K	INTERES: 3,5 sobre 5
CLASE DE PROGRAMA: Competición deportiva	REALISMO: 4,5 sobre 5

Manejo de la bicicleta, pero esperamos que el juego te guste y ganes rápidamente hasta el mismo **Perico Delgado...** ¡Buena suerte!



Mucho público en la edición de este Tour.

****

JUEGOS ESTIVALES

Es probable que ya sepáis que **Summer Games,** de **Epyx,** es uno de los escasos programas que han logrado mantener un cierto interés a través del tiempo, familiarizándonos con un tipo peculiar de *sprite.*

Con **Summer Games II** esta casa de *software* ha logrado lo que parecía imposible: Mejorar el nivel de sus juegos —si es que esto era posible—. En cualquier caso está claro que nos hallamos ante uno de los cinco mejores programas disponibles en el mercado internacional, hoy en día. Este Juego múltiple consta de ocho pruebas y dos pantallas de mero divertimento, que son las ceremonias de apertura y clausura. La primera escena es la misma que aparece en **Summer Games I,** aunque tomada en un ángulo ligeramente distinto. Después aparece el menú con las opciones para elegir una determinada prueba, la Olimpiada completa o las ceremonias. Volviendo a la clausura, aparece una novedad en su cuidada presentación: el hombre pájaro que asombró al público en las Olimpiadas de Los Angeles. Mientras, cae el sol en un bucólico atardecer. Llega la noche y el habitual dirigible visualiza, en *scroll,* una serie de mensajes, que ya verás por ti mismo.

El primer juego es el triple salto. Aparece el atleta listo para correr en el estadio lleno a rebosar de espectadores. Un truco simple para esta prueba consiste en mover la palanca del *joystick* hacia arriba/derecha, durante los dos primeros saltos y, en el último, a la izquierda. Para correr has de ir pulsando el disparador lo más rápidamente posible, para ganar velocidad (afortunados quienes tienen disparador automático). Si logras alcanzar una velocidad decente y sigues estas instrucciones alcanzarás los quince metros y medio de distancia, a partir de la cual se produce la repetición a cámara lenta en la parte superior de la pantalla (¡vaya chulada, tú!).

La siguiente prueba es sencilla, pero no por ello descuidada. Es la prueba de remo. Manejar la piragua es en cuestión mover la palanca de derecha a izquierda cada vez más rápidamente. Pero lo más efectivo demostró ser el movimiento «a lo loco». Hay que mencionar un detalle: el escenario de la prueba coincide con el ciclismo y es muy parecido a la prueba de carrera sobre patines de **Winter Games** (juegos de invierno).
El lanzamiento de jabalina es la prueba que habrás de realizar en siguiente lugar. En ella el sistema utilizado para correr es idéntico al empleado en el triple salto. Inclinando la palanca hacia arriba a la derecha harás que el atleta lance la jabalina, siendo proporcional el tiempo que mantengas la palanca inclinada al mayor o menor ángulo de lanzamiento. Se alcanza una mayor distancia cuando consigues que la jabalina no pase de la mitad de la pantalla. La cuarta prueba, la de equitación, es una de las más logradas. En ella debieron gastar mucho tiempo los programadores recomponiendo el salto, así como la carrera del caballo. Te hallarás inmerso en una carrera de obstáculos con un efecto de fondo muy real, gracias a los *scrollings*. Las caídas desde el caballo poseen un tremendo efecto de simulación de la realidad (casi te haces daño). Si pretendes saltar demasiado tarde el caballo no acepta y se queda «clavado» al pie de la barrera. La discusión entre quienes han conseguido ver este programa, antes de su (próxima) parición, en el mercado nacional, se centra en decidir si esta prueba está más lograda que la bajada por el río en *kayak*.
Si tu dedo continúa aún en forma, la esgrima es la siguiente prueba. Es recomendable adoptar una estrategia fundamentalmente defensiva. Ataca sin cesar y, sobre todo, da cuantos más golpes bajos puedas.
Una luz roja te indica cuándo sales fuera de la pista y la verde te dirá cuándo has tocado al adversario. Esta es una de las pruebas más difíciles de manejar.
Para finalizar, la prueba reina: bajada por las turbulentas aguas de un río en *kayak*. Realizada con una gusto exquisito, bajas un río plagado de corrientes, y deberás conducir a tu *kayak* entre los peligrosos remolinos, que te desvían continuamente. Los efectos han sido cuidados en sus mínimos detalles (incluso hay marcha atrás hacia ambos lados). Tras el final de la carrera se visualiza el recorrido efectuado, con las «puertas» pasadas con acierto, al revés y las «olvidadas», que serán con seguridad muchas la primera vez que te entrenes.



Con el estadio abarrotado de público tendrás que saltar, controlando el movimiento con el Joystick.



Si aún no estás muy cansado, puedes practicar esgrima. Tal vez es menos espectacular, pero sí la más difícil.

La prueba de equitación tiene un diseño elaborado, siendo el movimiento muy realista.



Uno de los juegos más espectaculares de este Summer Games II es el difícil descenso en kayak.



DATOS GENERALES	CALIFICACION
TITULO: Summer Games II	GRAFICOS:: 4,5 sobre 5
FABRICANTE: Epyx	COLOR: 4 sobre 5
ORDENADOR: Commodore 69	PRESENTACION: 5 sobre 5
MEMORIA: 64 K	INTERES: 4,5 sobre 5
CLASE DE PROGRAMA: Competición	REALISMO: 4,5 sobre 5

Revista de Software

MISION SUICIDA

Durante la Segunda Guerra Mundial se plantearon determinados objetivos, tales como ataques a poblaciones, zonas industriales, etc., y también, cómo no, ataques a presas. La segunda vía de agua más importante de Alemani era el río Oder, tanto en cuanto al tráfico fluvial que en él se llevaba a cabo como en la faceta de aprovechamiento del agua para fines industriales (siderurgia pesada...). Por tanto, el aniquilamiento de tales presas era un objetivo militar de suma importancia para los objetivos de los aliados.

Dam Buster recrea esa realidad histórica y nos devuelve a la situación en la que se vieron inmersos los tripulantes de los aviones **MARK III** durante los ataques que se llevaron a cabo en la presa mencionada.

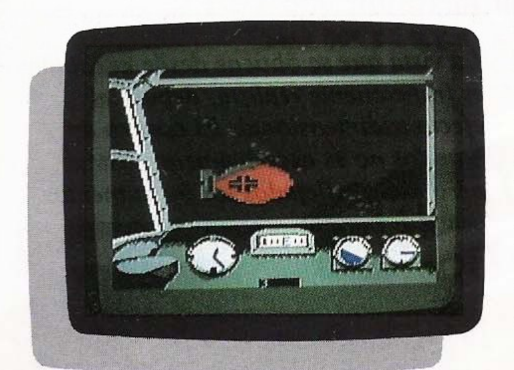
Antes de comentar el juego hay que insistir en la promoción que se dio en su presentación en Gran Bretaña: La **U. S. GOLD** (distribuidora del mejor *software* procedente de Estados Unidos en Europa) lo presentó como el juego más conseguido que distribuía hasta la fecha (no olvidéis que entre otros, ya comercializa el **Beach Head I** y **II, Zaxxon...**), y esto explica la gran expectación existente entre el público.

El juego permite tres modalidades de juego: I) Práctica de aproximación a la presa. II) Teniente de vuelo. III) Jefe de escuadrilla.

La primera opción te permite entrenarte en tu misión de bombardeo de la presa. Para entenderlo hay que explicar que la misión de los **Dam** era un tanto extraordinaria: tenían que lanzar una bomba especialmente diseñada, a una distancia, velocidad y altura de la presa determinada para que ésta rebotase encima de una malla protectora antitorpedos, alcanzase el muro de la presa y se hundiera por fin en el agua al pie del muro, estallando en este lugar para causar el mayor daño posible. Esta opción del juego te coloca en una posición muy cercana a la presa y sólo tendrás que entrenarte para la misión de lanzamiento de la bomba.

La segunda opción del juego (Teniente de vuelo) hará que empieces tu misión encima del Canal de la Mancha.

Tal y como sucedía en la realidad el avance de tu avión se verá dificultado por aviones alemanes **ME.110,** reflectores antiaéreos y globos. Cualquiera de estas tres amenazas tiene un antídoto de peso: una ametralladora del calibre 303; el juego consta de seis pantallas, que aparecen según indiques al ordenador dónde quieres colocarte en el avión: tienes cinco posiciones: piloto, artillero delantero, artillero trasero, navegante e ingeniero de vuelo. Además dispones de un mapa de Europa, que te indica la posición de tu avión, así como los diferentes atacantes que encontrarás en tu camino. A veces, mientras estás en posición de piloto, un avión enemigo realiza un ataque en tu flanco derecho. El ordenador te indicará esta situación de dos maneras. Por otra parte, oirás un sonido de disparos y, por otra, verás una casilla en la parte inferior de la pantalla en



Los globos alemanes dificultan el vuelo.

DATOS GENERALES	CALIFICACION
TITULO: The Dam Busters	GRAFICOS: 3,5 sobre 5
FABRICANTE: U. S. Gold	COLOR: 3,5 sobre 5
ORDENADOR: Commodore	PRESENTACION: 5 sobre 5
MEMORIA: 48 K	INTERES: 5 sobre 5
CLASE DE PROGRAMA: Reconstrucción histórica	REALISMO: 5 sobre 5

la que aparecerán números: éstos te indican que has de pasar a la pantalla a la que corresponden, por no existir un peligro inminente en la zona que representan.
La tercera opción de vuelo es muy similar a las dos anteriores, ya que la diferencia estriba en que tendrás que realizar la misión completa incluyendo el despegue.
Estamos ante un juego cuidado, tanto en los gráficos como en lo que se refiere a sonido, el ataque de los aviones recuerda vagamente al **Beach-Head,** aunque hay que reconocer que está mejor logrado. La mayor novedad del juego reside en la velocidad de cambio de las diferentes pantallas y, por ello, es fácil jugar durante horas y horas sin parar, el perfeccionamiento del juego alcanza incluso la posibilidad de realizar maniobras «sacarse» de encima los aviones enemigos.


El panel de instrumentos es muy realista.


Detalle del mapa del Continente.

★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★

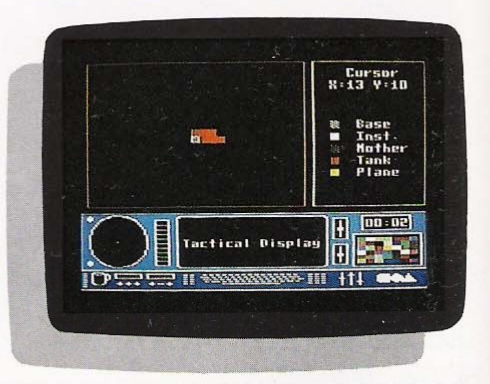
EL CAZA

Eres el piloto del famoso caza *Skyfox* y debes proteger las bases de la *Federación,* que son atacados por los tanques y la aviación enemigos; procedentes de su nave nodriza.
Te introduces en la sofisticada máquina de guerra, presionas el botón y sales disparado. Ganas cada vez más altura, a medida que llevas la palanca hacia ti.
Tus cánones láser son disparados con el botón del *joystick,* y la visión tridimensional, que obtienes desde la carlinga de tu nave, te permite ver cómo alcanzas al invasor.
El piloto automático, activado mediante las teclas f7 o A, te facilita la localización del enemigo más próximo y dirigirte hacia él para interceptarlo. También puedes utilizar el *display* táctico para ubicarlos.
En la pantalla del radar obtienes una vista aérea de tu nave, en el centro, y de los enemigos más cercanos a ti.
Dis pones de indicador de combustible, radar de lago alcance, indicador de altitud y misiles inteligentes.
El juego se divide en dos partes. En la primera debes eliminar a los tanques y en la segunda, que tiene más que ver con un simulador de vuelo, te enfrentas a los aviones enemigos.



La vista que se obtiene desde la cabina es francamente realista, debido al efecto tridimensional. El cuadro de mandos no es excesivamente vistoso, pero están reflejados los instrumentos precisos para la «caza».





DATOS GENERALES	CALIFICACION
TITULO: Skyfox	GRAFICOS: 4 sobre 5
FABRICANTE: Ariola soft	COLOR: 4 sobre 5
ORDENADOR: C-64	PRESENTACION: 3,5 sobre 5
MEMORIA: 64 K	INTERES: 4,5 sobre 5
CLASE DE PROGRAMA: Defensa aérea	REALISMO: 4,5 sobre 5

Elidon es Fantasía ven a Recogerla



Elidon puede ser tuyo. Si te suscribes *ahora* a la excelente revista Commodore Soft Magazine recibirás *completamente gratis* el juego Elidon, cuyo precio en el mercado es de 1.800 pesetas.

Si eres un fanático de las aventuras, Elidon es para ti. **Este juego es el nº 1 en Inglaterra,** mostrando un despliegue de gráficos soberbios y un grado de dificultad pocas veces superado. Deberás conducir el hada a través del bosque de Elidon en busca de las pociones necesarias para que florezcan las plantas.



BOLETIN DE SUSCRIPCION

MONSER, S.A. - COMMODORE SOFT MAGAZINE - c/ Argos, 9 - 28037 Madrid - Tlf. 742 72 12 / 96

Deseo suscribirme por 12 ejemplares y 12 cassettes de commodore soft Magazine por 5.940 ptas. a partir del número____

El importe lo haré efectivo:

- ☐ Por giro postal n.º
- ☐ Por contra reembolso a la recepción del primer ejemplar
- ☐ Por talón bancario a Monser, S.A.
  Gastos de envío incluidos.

Nombre y apellidos ....

Domicilio .... Tlf. .... Ciudad .... C.P. ....

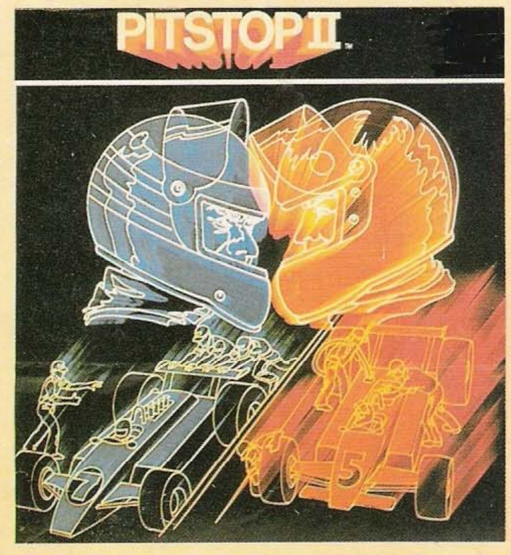
Fecha .... Firma



Nº 1 EN U.S.A. CON

PARA COMMODORE 64, PROXIMAMENTE PARA SPECTRUM





PITSTOP II

- 6 circuitos diferentes.
- Excelentes gráficos.
- Escucha el rugir de los motores.
- Doble acción simultánea.

Con PITSTOP II puedes experimentar la emoción de la velocidad y el riesgo a medida que luchas con tu oponente en una carrera contra el tiempo.

Es, sin duda, el mejor juego de carreras existente.

- 1 ó 2 Jugadores • Joystick • **P.V.P.: 2.300 ptas.**

IMPOSSIBLE MISSION

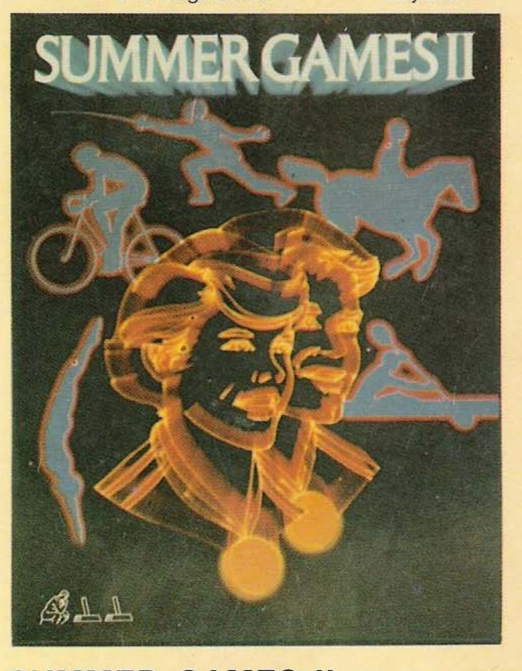
- Soberbios Gráficos.
- Voz Sintetizada.
- Concepto original del juego.
- Juego del año 1985.

Son algunas de sus características.

Como miembro del escuadrón antiterrorista computerizado, tu misión es encontrar y alcanzar al infame ELVIN. Debes encontrar un paso a través de las habitaciones y túneles de su Cuartel General, tratando de evitar los robots protectores.

Tu misión es salvar el mundo, pero date prisa.

- 1 Jugador • Joystick • **P.V.P.: 2.300 ptas.**



SUMMER GAMES II

No es tarde para prepararse cara a 1988, con la dieta correcta, entrenamiento adecuado y horas de práctica lo puedes conseguir.
Mientras tanto ponte el chandal, coge el joystick y deja que SUMMER GAMES II te dé ocho nuevas formas de obtener el ORO.
Ciclismo, Vallas, Kayak, Triple salto, Salto de altura, Jabalina, Hípica y Esgrima son las pruebas.
Entrega de medallas al final de cada prueba, Ceremonia de apertura y cierre, son algunas de sus características.

- Hasta 8 Jugadores • Joystick • **P.V.P.: 2.300 ptas.**



BREAKDANCE

La locura del momento, el BREAKDANCE, no debes perdértela.

Ahora cualquiera puede bailar BREAKDANCE, coge el joystick y controla tu bailarín en moon walking, stretching, popping y breaking.

Elige entre cuatro modos de juego: Un juego de acción en que tu bailarín lucha con las bandas de BREAKERS; Intenta seguir el ritmo de tu oponente; Crea tu propia coreografía y el ordenador te hará un playback de ella; Estudia la secuencia de pasos para realizar los distintos movimientos BREAK.

Aprende a bailar BREAKDANCE hoy, EPYX te lo hace fácil.

- 1 ó 2 Jugadores • Joystick • **P.V.P.: 2.300 ptas.**

Solicítelo a su distribuidor habitual o a:
COMPULOGICAL, S. A.
Santa Cruz de Marcenado, 31
28015 MADRID
Teléfono 241 10 63